



Architectural exploration and performance analysis of Vertically-Partially-Connected Mesh-based 3D-NoC

Maryam Bahmani

► To cite this version:

Maryam Bahmani. Architectural exploration and performance analysis of Vertically-Partially-Connected Mesh-based 3D-NoC. Other [cs.OH]. Université de Grenoble, 2013. English. NNT : 2013GRENM066 . tel-01070020v2

HAL Id: tel-01070020

<https://theses.hal.science/tel-01070020v2>

Submitted on 25 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

« **Maryam BAHMANI** »

Thèse dirigée par « **Maryam BAHMANI** » et
codirigée par

« **Dr. Abbas SHEIBANYRAD** » et « **Prof. Frédéric PETROT** »

préparée au sein du **Laboratoire TIMA**

dans l'École Doctorale **Mathématiques, Sciences et Technologies de
l'Information, Informatique (MSTII)**

Exploration architecturale et étude des performances des réseaux sur puce 3D partiellement connectés verticalement

Thèse soutenue publiquement le « 9 décembre 2013 »,
devant le jury composé de :

Prof. Ian O'CONNOR

Professeur, ECL France, rôle (Rapporteur)

Dr. Gilles SASSATELLI

Directeur de recherche, LIRMM France, rôle (Rapporteur)

Dr. Jean-Philippe DIGUET

Directeur de recherche, Lab-STICC France, rôle (président du jury)

Dr. Franck WAJSBURT

Maître de Conférences, LIP6 France, rôle (Membre)

Dr. Abbas SHEIBANYRAD

Chargé de recherche, TIMA/SLS France, rôle (Encadrant de la thèse)

Prof. Frédéric PETROT

Professeur, TIMA/SLS France, rôle (Directeur de la thèse)



L'utilisation de la troisième dimension peut entraîner une réduction significative de la puissance et de la latence moyenne du trafic dans les réseaux sur puce (NoC¹). La technologie des vias à travers le substrat (ou TSV²) est la technologie la plus prometteuse pour l'intégration 3D, car elle offre des liens verticaux courts qui remédient au problème des longs fils dans les NoCs-2D. Les TSVs sont cependant énormes et les processus de fabrication sont immatures, ce qui réduit le rendement des systèmes sur puce à base de NoC-3D. Par conséquent, l'idée de réseaux sur puce 3D partiellement connectés verticalement a été introduite pour bénéficier de la technologie 3D tout en conservant un haut rendement. En outre, de tels réseaux sont flexibles, car le nombre, l'emplacement et l'affectation des liens verticaux dans chaque couche peuvent être décidés en fonction des exigences de l'application.

Cependant, ce type de réseaux pose un certain nombre de défis :

Le routage est le problème majeur, car l'élimination de certains liens verticaux fait que l'on ne peut utiliser les algorithmes classiques qui suivent l'ordre des dimensions. Pour répondre à cette question nous expliquons et évaluons un algorithme de routage déterministe appelé "Elevator First", qui garanti d'une part que si un chemin existe, alors on le trouve, et que d'autre part il n'y aura pas d'interblocages.

Fondamentalement, la performance du NoC est affecté par a) la micro architecture des routeurs et b) l'architecture d'interconnexion.

L'architecture du routeur a un effet significatif sur la performance du NoC, à cause de la latence qu'il induit. Nous présentons la conception et la mise en œuvre de la micro-architecture d'un routeur à faible latence implantant l'algorithme de routage Elevator First, qui consomme une quantité raisonnable de surface et de puissance.

Du point de vue de l'architecture, le nombre et le placement des liens verticaux ont un rôle important dans la performance des réseaux 3D partiellement connectés verticalement, car ils affectent le nombre moyen de sauts et le taux d'utilisation des FIFOs dans le réseau. En outre, l'affectation des liens verticaux vers les routeurs qui n'ont pas de ports vers le haut ou/et le bas est une question importante qui influe fortement sur les performances. Par conséquent, l'exploration architecturale des réseaux sur puce 3D partiellement connectés verticalement est importante. Nous définissons, étudions et évaluons des paramètres qui décrivent le comportement du réseau, de manière à déterminer le placement et l'affectation des liens verticaux dans les couches de manière simple et efficace. Nous proposons une méthode d'estimation quadratique visant à anticiper le seuil de saturation basée sur ces paramètres.

¹ Network-on-Chip

² Through-Silicon Via

Abstract

Utilization of the third dimension can lead to a significant reduction in power and average hop-count in Networks-on-Chip (NoC). TSV technology, as the most promising technology in 3D integration, offers short and fast vertical links which copes with the long wire problem in 2D NoCs. Nonetheless, TSVs are huge and their manufacturing process is still immature, which reduces the yield of 3D NoC based SoC. Therefore, Vertically-Partially-Connected 3D-NoC has been introduced to benefit from both 3D technology and high yield. Moreover, Vertically-Partially-Connected 3D-NoC is flexible, due to the fact that the number, placement, and assignment of the vertical links in each layer can be decided based on the limitations and requirements of the design. However, there are challenges to present a feasible and high-performance Vertically-Partially-Connected Mesh-based 3D-NoC due to the removed vertical links between the layers.

This thesis addresses the challenges of Vertically-Partially-Connected Mesh-based 3D-NoC:

Routing is the major problem of the Vertically-Partially-Connected 3D-NoC. Since some vertical links are removed, some of the routers do not have up or/and down ports. Therefore, there should be a path to send a packet to upper or lower layer which obviously has to be determined by a routing algorithm. The suggested paths should not cause deadlock through the network. To cope with this problem we explain and evaluate a deadlock- and livelock-free routing algorithm called Elevator First.

Fundamentally, the NoC performance is affected by both 1) micro-architecture of routers and 2) architecture of interconnection.

The router architecture has a significant effect on the performance of NoC, as it is a part of transportation delay. Therefore, the simplicity and efficiency of the design of NoC router micro architecture are the critical issues, especially in Vertically-Partially-Connected 3D-NoC which has already suffered from high average latency due to some removed vertical links. Therefore, we present the design and implementation the micro-architecture of a router which not only exactly and quickly transfers the packets based on the Elevator First routing algorithm, but it also consumes a reasonable amount of area and power.

From the architecture point of view, the number and placement of vertical links have a key role in the performance of the Vertically-Partially-Connected Mesh-based 3D-NoC, since they affect the average hop-count and link and buffer utilization in the network. Furthermore, the assignment of the vertical links to the routers which do not have up or/and down port(s) is an important issue which influences the performance of the 3D routers. Therefore, the architectural exploration of Vertically-Partially-Connected Mesh-based 3D-NoC is both important and non-trivial. We define, study, and evaluate the parameters which describe the behavior of the network. The parameters can be helpful to place and assign the vertical links in the layers effectively. Finally, we propose a quadratic-based estimation method to anticipate the saturation threshold of the network's average latency.

Acknowledgment

Firstly, I would like to express my sincere gratitude to my supervisor Prof. Frédéric PETROT, and advisor Dr. Abbas SHEIBANYRAD for their continuous support of my thesis studies and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me throughout the execution of this research, and the writing of this thesis.

Besides my supervisor and advisor, I would like to thank the rest of my thesis committee: Prof. Ian O’CONNOR, Dr. Gilles SASSATELLI, Dr. Jean-Philippe DIGUET, and Dr. Franck WAJSBURT for accepting our invitation to evaluate and examine this thesis.

I would also like to extend my appreciation to my colleagues and friends in the TIMA laboratory: Prof. Frédéric ROUSSEAU, Olivier MULLER, Stéphane MANCINI, Nicolas FOURNEL, Sahar FOROUTAN, Pierre-Henri HORREIN, Damien HEDDE, Florentine DUBOIS, Marcos CUNHA, Clément DESCHAMPS, Adrien PROST-BOUCLE, Yan XU, Ashraf ELANTABLY, Sofiane LAGRAA, Luc MICHEL, Guillaume SARRAZIN, Panagiota PAPAVERAMIDOU, Fabien CHAIX, Vladimir PASCA, Wassim MANSOUR, Fatemeh JAVAHERI, Sahar JAVDANI, Thomas DUPORT, and Dany and Roland VIALY for their kindness, support, and all the fun we have had during the last three years.

Also I thank the administration team in the TIMA laboratory: Prof. Dominique BORRIONE, Laurence BEN TITO, Anne-Laure FOURNERET-ITIE, Sophie MARTINEAU, Marie-Christine SALIZZONI, Youness RAJAB, Frederic CHEVROT, and Nicolas GARNIER for supporting and helping me kindly.

I would like to express my deepest gratitude to my family and especially to my mother, Nazafarin GHAHRAMANI and my brother, Khashayar BAHMANI for supporting me spiritually throughout my life; without their encouragement and support, I would not have a chance to be in TIMA laboratory in Grenoble.

Last but not the least; I would also like to extend my appreciation to my beloved, Dr. Gareth KEELEY during the last three years for his unconditional love, kindness, and support.

Table of Contents

Résumé (Français)

Chapitre Un: Introduction.....	1
Chapitre Deux: Definition Des Problemes	5
Chapitre Trois: Etat De L'art	11
Chapitre Quatre: Algorithm De Routage Elevator First.....	15
Chapitre Cinq: Elevator First Routeur Microarchitecture	21
Chapitre Six: Exploration De L'architecture De Reseau Sur Puce 3d Partiellement Connectés Verticalement.....	25
Chapitre Sept: Conclusion Et Perspectives	37

The Main Part (La Partie Principale (Anglais))

Chapter One: Introduction	41
Chapter Two: Problem Definition	45
1. System-on-Chip (SoC)	45
1.1. Shared Medium and Crossbar Interconnection	46
1.2. Network-on-Chip (NoC).....	47
2. Three Dimensional NoC (3D NoC).....	47
2.1. Dense Integration	47
2.2. Lower Average Latency and Power Consumption	47
2.3. Flexible Integration and Feasible On-Chip Memory	48
3. Through-Silicon-Vias (TSVs)	48
3.1. TSV Area Consumption.....	49
3.2. Yield vs. Number Of TSVs.....	49
4. Vertically-Partially-Connected 3D-NoC	50
5. Issues in Vertically-Partially-Connected 3D-NoC	51
5.1. Problem 1: Routing in Vertically-Partially-Connected 3D-NoC	52
5.2. Problem 2: Performance Degradation in Vertically-Partially-Connected 3D-NoC	53
5.2.1. Micro-Architecture Level Analysis.....	54
5.2.2. System Level Analysis.....	54
6. Conclusion	56
Chapter Three: State Of The Art.....	59
1. 3D NoC Routing Algorithm Related Works	61
2. 3D NoC Router Micro-Architecture Related Works	66
3. 3D NoC Architectural Exploration Related Works	68

4. Conclusion	71
Chapter Four: Elevator First Routing Algorithm	73
1. Elevator Concept	73
2. Elevator First Routing Algorithm	75
2.1. Deadlock in Elevator First Routing Algorithm	76
3. Solution for Deadlock in Elevator First Routing Algorithm	76
4. Elevator First Routing Algorithm with Z+ and Z- Virtual Networks	77
5. Experimental Results of Elevator First Routing Algorithm	80
5.1. Average Hop-Count	81
6. Conclusion	82
Chapter Five: Micro-Architecture Of Elevator First Router	83
1. DSPIN Router	83
2. Elevator First Router	85
3. Addition and Removal of Temporary Header in Elevator First Router	86
4. The Micro-Architecture of the Ports in Elevator First Router	88
5. RTL-Level Synthesis	92
6. Conclusion	93
Chapter Six: Architectural Exploration Of Vertically-Partially-Connected 3d-Noc	95
1. Analysis of Number of Possible Topologies in Vertically-Partially-Connected 3D-NoC	98
2. Region Concept	98
3. Pattern-based Vertically-Partially-Connected Mesh-based 3D-NoC Placement and Assignment Algorithm	100
4. Uniform Elevator Assignment (UEA)	106
4.1. Experimental Results of Uniform Elevator Assignment Method	107
5. Quadratic-Based Estimation Method	109
5.1. The Parameters in Matrix of Parameters [Param]	111
5.2. Experimental Results of Quadratic-based Estimation Method	112
6. Conclusion	114
Chapter Seven: Conclusion and Future Works	119
Appendix A: Ideal Region Hop-Count	123
Acronym	125
References	127

List of Figures

Figure 1.1: The thesis roadmap.....	42
Figure 2.1: Computation-centric vs. communication-centric approach.....	45
Figure 2.2: Heterogeneous and homogeneous Multi-Processor systems.....	46
Figure 2.3: Scalable Evolution of System on Chip interconnections (from bus based SoC to NoC based SoC)	46
Figure 2.4: Dimensional Evolution of System on Chip interconnections (from 2D NoC based SoC to 3D NoC based SoC)	47
Figure 2.5: Off-chip links vs. On-chip links	48
Figure 2.6: TSV with landing pad and keep-out area is several times larger than gates and memory cells [35]	49
Figure 2.7: Yield vs. TSV count [45]	50
Figure 2.8: a) An example of Vertically-Partially-Connected 3D-NoC, b) NoC routers category in Vertically-Partially-Connected 3D-NoC	51
Figure 2.9: Vertically-Partially-Connected 3D-NoC challenges	52
Figure 2.10: Deadlock-free Dimension order Z-First routing algorithm for Fully-Connected Mesh-based 3D-NoC	53
Figure 2.11: Routing problem in Vertically-Partially-Connected 3D-NoC.....	53
Figure 2.12: Transport latency	54
Figure 2.13: The number of possible topologies (T) for N equals to 9, 16, 25 and 36 presented in logarithm scale	56
Figure 2.14: An example of possible placements and assignments for two 3D nodes and four nodes in a layer	55
Figure 3.1: Various NoC topologies (not to scale). (a) 2-D IC–2-D NoC. (b) 2-D IC–3-D NoC. (c) 3-D IC–2-D NoC. (d) 3-D IC–3-D NoC [43].....	60
Figure 3.2: Examples of XNoTs topologies (a-d), proposed routing and a generic XNoT architecture [48].....	62
Figure 3.3: Side view of 3D NoC with the dTDMA bus[49]	63
Figure 3.4: Example of 7 different possible paths in AdaptiveZ routing[49].....	63
Figure 3.5: The MXYZ routing [51].....	64
Figure 3.6: An irregular region oriented multicasting framework [51]	64
Figure 4.1: X-First Routing Algorithm as intra layer routing mechanism	74
Figure 4.2: Elevator types	74

Figure 4.3: Elevator First Routing Algorithm.....	75
Figure 4.4: Deadlock in Elevator First Routing Algorithm	76
Figure 4.5: Elevator First Routing Algorithm with Z+ and Z- Virtual Networks	77
Figure 4.6: The block diagram of Elevator First Router.....	78
Figure 4.7: Elevator First Routing in LOCAL input port	78
Figure 4.8: Elevator First Routing in 2D input ports	79
Figure 4.9: Elevator First Routing in 3D input ports	79
Figure 4.10: Average latency using uniform traffic distribution	81
Figure 4.11: Average latency using localized traffic distribution.....	81
Figure 4.12: Average Hop-Count using uniform traffic	82
Figure 4.13: Average Hop-Count using localized traffic.....	82
Figure 5.1: DSPIN router [73]	83
Figure 5.2: Ports handshaking in conventional NoC router.....	84
Figure 5.3: Elevator-First Router.....	85
Figure 5.4: Addition and removal of the temporary headers in Elevator First routing algorithm.....	87
Figure 5.5: a) Addition the temporary header b) Removal the temporary header	88
Figure 5.6: LOCAL Elevator First Finite State Machine	89
Figure 5.7: Micro-architecture of LOCAL Elevator First input port.....	90
Figure 5.8: UP_Elevator First input port	91
Figure 5.9: DOWN_Elevator First input port.....	91
Figure 5.10: 2D Elevator First input port.....	92
Figure 5.11: 2D Elevator First Finite State Machine.....	92
Figure 5.12: Router Area Comparison.....	93
Figure 6.1: Number of vertical links placements in a Mesh-based layer.....	96
Figure 6.2: Number of 2D nodes assignments in a Mesh-based layer.....	96
Figure 6.3: Number of topologies in Vertically-Partially-Connected Mesh-based 3D-NoC ..	96
Figure 6.4: The gap between the best and worst saturation thresholds of 18.5 K topologies for different number of Elevators in 6×6×5 network	97
Figure 6.5: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in 5×5×5 network	97

Figure 6.6: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in $4 \times 4 \times 5$ network	97
Figure 6.7: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in $3 \times 3 \times 5$ network	97
Figure 6.8: Region concept in Vertically-Partially-Connected Mesh-based 3D-NoC	99
Figure 6.9: The Load & Total Region-Degree	100
Figure 6.10: Pattern-based Placement and Assignment in Vertically-Partially-Connected 3D-NoC.....	101
Figure 6.11: Expandability of Pattern-based Placement in Vertically-Partially-Connected 3D-NoC.....	102
Figure 6.12: The initialization phase of the pattern-based algorithm	103
Figure 6.13: The pseudo code of the pattern-based placement and assignment algorithm ...	104
Figure 6.14: Three possible placement and assignment of Elevators in 5×5 Mesh layer.....	106
Figure 6.15: The comparison between the Uniform Elevator Assignment (UEA) and random topologies (placements and assignment) in $(5 \times 5 \times 5)$ network.....	108
Figure 6.16: The flow of quadratic-based estimation method	110
Figure 6.17: The proportion of prioritized-based selection	113
Figure 6.18: Average and standard deviation of errors when $\mathbf{NZ} = 6 \times 6 \times 5$	115
Figure 6.19: Average and standard deviation of errors when $\mathbf{NZ} = 5 \times 5 \times 5$	115
Figure 6.20: Average and standard deviation of errors when $\mathbf{NZ} = 4 \times 4 \times 5$	115
Figure 6.21: Average and standard deviation of errors when $\mathbf{NZ} = 3 \times 3 \times 5$	115
Figure A: B is the sum of hop-count between A number of 2D nodes and their Elevator	115

List of Tables

Table 3.1 min-max impact on the energy, latency and area values of vertically partially 3D NoC that were obtained are compared to the ones of the 3D full vertically interconnected NoC in [55] and [56]	69
Table 3.2: Summary of the presented related works; NA: Not Applicable	70
Table 5.1: The connections of Z + and Z- Switch Fabrics.....	86
Table 5.2: Area of different types of Elevator-First Router.....	93
Table 6.1: The results of the pattern-based placement and assignment algorithm for a 5×5 Mesh layer when HP equals ‘1’	105
Table 6.2: The architectural parameters of Vertically-Partially-Connected 3D-NoC	112

CHAPITRE UN: INTRODUCTION

Réseau sur puce (NoC) a été introduit initialement par Guerrier et al. [14] pour faire face à la question de l'évolutivité dans les interconnexions (Shared Medium) au sein de System-on-Chip (SoC), dans lequel le nombre de la propriété intellectuelle (IP) des blocs a été fait qu'augmenter. Certains des avantages de NoC sont comme suit:

Evolutivité: En raison de l'existence de routeurs NoC, grand nombre d'adresses IP peut communiquer par eux sans aucune congestion et avec une consommation d'énergie plus efficace. En outre, les liaisons de la bande passante agrégées s'adaptent à la taille du réseau.

Flexibilité: La conception de routeurs NoC peut être utilisé pour toutes les tailles de réseau et ré-instancié pour de nombreuses technologies. En outre, les blocs IP peuvent être choisis parmi les différentes applications (hétérogène NoC) ou similaires (homogène NoC) et travaillent avec différents ou similaires fréquences d'horloge [13-17]. En outre, le réseau peut être à commutation de paquets ou à commutation de circuits. Bien sûr, la première méthode se termine dans le cadre plus simultanée et moins de consommation d'énergie et, cette dernière méthode apporte davantage de ressources pour les applications sensibles à la latence.

Transparence: Les couches atomiques et indépendants des flux de conception de réseau sur puce conduisent à la réduction du (Time-to-Market) depuis l'ajout de nouveaux services à une seule couche peut être nécessaire de modifier les fonctionnalités à une couche, en réutilisant les fonctions prévues à toutes les autres couches [38].

Modularité: Les décisions de routage des paquets à travers le réseau d'interconnexion si peuvent être faites par les routeurs NoC de manière distribuée ou par une unité de fabrication dans le réseau central décision. Dans le premier, non seulement de nombreux chemins peuvent être utilisés entre les nœuds de source et de destination en même temps, mais aussi tous les liens ne sont pas occupés quand ils ne sont pas nécessaires. L'ancienne extrémité de conception robuste et tolérante faute et ce dernier conduit à une moindre consommation d'énergie.

D'autre part, le réseau sur puce souffre du retard et une grande consommation d'énergie de longs fils et la croissance de nombre de sauts lorsque le nombre de cœurs IP grandit [44]. L'utilisation de la troisième dimension (technologie 3D) peut conduire à une réduction significative de la puissance et moyenne nombre de sauts dans les réseaux sur puce (NoC) [39]. La combinaison de la technologie et de la NoC approche 3D se termine dans l'intégration de différentes couches denses sur puce avec différentes technologies telles que les systèmes sur puce sur puce mémoire [42]. En outre, la technologie TSV, comme la technologie la plus prometteuse en matière d'intégration 3D, offre des liens verticaux courts et rapides, entre couches empilées [28]. Néanmoins, TSV sont énormes [35] et de leur processus de fabrication est encore immature, ce qui réduit le rendement [45] de SoC à base de NoC 3D.

Par conséquent, de réseaux sur puce 3D partiellement connectés verticalement a été introduit pour cibler à la fois en trois dimensions (3D) et de la technologie à haut rendement. En outre, de réseaux sur puce 3D partiellement connectés verticalement est flexible, en raison du fait que le nombre, l'emplacement et l'affectation des liens verticaux dans chaque couche peuvent être décidé reposent sur les exigences de la conception. Cependant, il ya des défis à présenter un réseau sur puce 3D partiellement connectés verticalement possible et de haute performance en raison des liens verticaux supprimés entre les couches.

Par conséquent, cette thèse aborde les défis suivants de réseaux sur puce 3D partiellement connectés verticalement:

A. L'algorithme de routage

Le routage est le problème majeur de réseaux sur puce 3D partiellement connectés verticalement. Étant donné que certaines liaisons verticales sont éliminés, quelques-uns des routeurs n'ont pas vers le haut ou/et vers les bas ports. Par conséquent, il devrait y avoir un chemin d'accès pour envoyer un paquet à la couche supérieure ou inférieure, qui doit évidemment être déterminé par un algorithme de calcul d'itinéraire. Les chemins proposés ne doivent pas provoquer un blocage à travers le réseau. Par conséquent, dans une impasse et algorithme de routage deadlock-free and livelock-free est proposé et évalué pour de réseaux sur puce 3D partiellement connectés verticalement. Ce problème a été abordé dans [68] et dans le chapitre quatre de cette thèse.

B. La dégradation des performances

La performance NoC est affectée par 1) microarchitecture de routeurs et 2) l'architecture d'interconnexion :

B.1. L'amélioration des performances au niveau microarchitecture

Le routeur microarchitecture a un effet significatif sur la performance de NoC, car il fait partie de retard de transport de paquets. Par conséquent, la simplicité et l'efficacité de la conception de NoC routeur micro architecture sont des questions cruciales, en particulier dans réseaux sur puce 3D partiellement connectés verticalement qui a déjà souffert de haute latence moyenne en raison de certains liens verticaux qui manquent. Par conséquent, non

seulement le routeur doit transmettre exactement et rapidement les paquets en fonction de l'algorithme de calcul d'itinéraire, mais il convient également de consommer une quantité raisonnable de la zone et le pouvoir. Ce problème a été abordé dans [72] et dans le chapitre cinq de cette thèse.

B.2. L'amélioration des performances au niveau de l'architecture

Le nombre et le placement des liens verticaux ont un rôle clé dans l'exécution de la de réseaux sur puce 3D partiellement connectés verticalement, car ils affectent le nombre de sauts moyenne et lien et tampon utilisation dans le réseau. En outre, la cession des liens verticaux aux routeurs qui n'ont pas ou/et vers le bas le port(s) est une question importante qui influe sur la performance des routeurs 3D grandement. Par conséquent, l'exploration architecturale de réseaux sur puce 3D partiellement connectés verticalement est importante et non négligeable. Définition, l'étude et l'évaluation des paramètres qui affectent le comportement du réseau peut être utile de placer et d'attribuer les liens verticaux dans les couches. Enfin, en proposant une méthode qui estime rapidement le seuil de saturation de la latence moyenne du réseau sur la base de quelques paramètres est intéressant et pratique. Ce problème a été abordé dans le chapitre six de cette thèse.

CHAPITRE DEUX:

DEFINITION DES PROBLEMES

L'émergence de la troisième dimension dans SoC interconnexion est le résultat de la mise à l'échelle géométrique dans plateformes système-sur-puce (SoC). La technologie 3D peut maintenir la croissance de la performance d'IC (integrated circuits) et le nombre de transistors.

Through-Silicon-Vias (TSVs) est une technologie de niveau de la tranche la plus dense, pratique et prometteur de liens verticaux dans inter-gradations 3D. C'est une réalisation très utile qui permet de réduire considérablement le temps de latence d'interconnexion et augmente la flexibilité de conception. En outre, TSV sont 50 fois plus court et 10 fois plus rapide que les fils intra couche [29], qui ont tous des caractéristiques décrites finissent aussi dans une moindre consommation d'énergie. Bien que, la 3D est une méthode d'intégration viable, il ya des défis tels que la taille TSV et la relation de rapport de rendement et le nombre de TSV.

Les désavantages de TSV

L'un des défis de l'intégration 3D est la surface de la garniture de TSV. Comme il est montré dans [35] et [24], la piste d'atterrissage de TSV et garder-hors zone est considérablement grande. Habituellement, des circuits supplémentaires sont nécessaires pour la protection de décharge électrostatique et de l'adaptation de niveau de signal peuvent également être nécessaires. Selon ce document, en utilisant de plus de TSV, bien que le délai réduise de manière significative, la consommation de la zone d'interconnexion augmente considérablement. Par conséquent, non seulement moins d'espace libre sur chaque couche reste logique et fils, mais aussi la longueur des fils horizontaux augmente.

En outre, la précision d'alignement de TSV dans les différentes couches est très importante. Bien TSV placement sont très efficaces pour la synchronisation et les questions de pouvoir, de fabrication de TSV est complexe du point de vue de la technologie et pourrait ne pas être facile à fabriquer. C'est la raison pour laquelle même si le placement régulier de TSV n'est pas très efficace, il est parfois plus économique et a faible temps d'accès au marché pour les utiliser contre le placement irrégulier. Dans l'ensemble, la mise en place de TSV est une question importante.

En autre problème important dans l'intégration 3D est le nombre de TSV. Le (yield ratio) est réduite par l'augmentation du nombre de TSV. Bien entendu, la valeur de rendement de la réduction de la proportion de diverses technologies est différente. Il est basé sur la sensibilité et les caractéristiques de la technologie de fabrication TSV. Cependant, dans toutes les technologies donner baisse (défauts se lève) de manière exponentielle lorsque le nombre de TSV est supérieur à 1000 (IMEC) ou 10000 (IBM et HRI- JP).

De même que des estimations en ITRS 2009 [36], le processus de conception et les outils de CAD (Computer Aided Design) de puce en trois dimensions d'empilage grâce à l'utilisation de haute densité TSV est un domaine clé pour répondre latence et de la puissance des préoccupations et seront améliorées. Néanmoins, l'intégration 3D peut être plus utile quand un équilibre entre le nombre raisonnable de liens énormes et peu fiables verticales (TSV) et les avantages de leur ont été trouvée. En d'autres termes, non seulement nous devons profiter de TSV courts, rapides et flexibles, mais aussi nous devons avoir un rendement acceptable et une superficie suffisante pour la logique de la couche intra et l'interconnexion.

Réseau sur puce 3D partiellement connectés verticalement

Réseau sur puce 3D partiellement connectés verticalement est à base de SoC empilés. Dans cette architecture les routeurs NoC sont divisés en deux catégories: a) 2D Routeurs et b) Routeurs 3D. Routeurs 2D sont les ports classiques 5 routeurs qui ne sont pas en mesure de recevoir ou envoyer des paquets à partir de/à la baisse ou couches supérieures (N, W, S, E, L). Alors que, routeurs 3D ont deux ports haut et bas qui leur permettent de recevoir et d'envoyer des paquets à partir de/à traverser les couches supérieures et/ou inférieures, fonction de leur type. Routeurs 3D dans Réseau sur puce 3D partiellement connectés verticalement sont classés en trois catégories: a) les routeurs 3D qui ont seulement UP port appelé "3D routeur - UP", b) des routeurs 3D qui ont seulement BAS port appelé " 3D routeur - DWN " et c) des routeurs 3D qui ont deux ports HAUT et BAS appelés "3D Router-UP/DWN ". En outre, les flux de données dans les liens verticaux dans Réseau sur puce 3D partiellement connectés verticalement peut être différent. Bien sûr, cela ajoute plus de flexibilité à la Réseau sur puce 3D partiellement connectés verticalement et le rend plus possible et être fabriqué. Par conséquent, nous pouvons conclure que la Réseau sur puce 3D partiellement connectés verticalement présente les caractéristiques suivantes:

- Taille:

Nombre de nœuds dans chaque couche peut être différente.

- Nombre, de placement et de cession de TSV:

Nombre et placement du TSV entre deux couches adjacentes peuvent être différents. Affectation des TSV aux noeuds 2D à l'intérieur d'une couche donnée n'a pas de contrainte technologique et est donc arbitraire.

Les désavantages de Réseau sur puce 3D partiellement connectés verticalement

Réseau sur puce 3D partiellement connectés verticalement peut créer une infrastructure très dense, souple, et être fabriqué pour SoC, des défis existent encore en raison des TSV supprimé:

- Routage dans Verticalement-partiellement-Connecté 3D NoC
- La dégradation des performances due à la réduction de la largeur de bande verticale

Dans cette thèse, nous avons classé les problèmes basés sur le système et les points de vue architectural. Dans [2], réseau sur puce a été divisé en deux parties principales de l'architecture et du système. Comme intéressant mentionné dans [2], non seulement ces deux catégories couvrent toutes les questions d'architecture de NoC, mais aussi qu'ils se complètent mutuellement:

- Niveau Système:
Cette partie contient la topologie et de routage mécanisme et couvre l'infrastructure des nœuds et les protocoles de connectivité entre eux.
- Niveau Architecture:
Cette partie comprend RTL de routeurs architectures et des interfaces réseau (NI) et couvre les éléments détaillés dans un nœud de réseau.

Comme mentionné précédemment, l'interconnexion joue un rôle clé dans la NoC. Depuis, "différentes interconnexions" apporter "différents algorithmes de routage de paquets" qui conduisent à "différents routeur microarchitectures". Donc, ils sont étroitement tricotées les unes aux autres. Connectivité correspondante NoC (paquet de routage), la performance (latence réseau), et le coût (superficie et la consommation d'énergie) sont sensible affecté par l'autre. De plus, les algorithmes de routage complexes peuvent minimiser les sauts dans les chemins entre les nœuds et éviter arguments dans les liens. Ces complexités de routage fin à plus de logiques de commande à microarchitectures routeur qui conduisent finalement à une plus grande surface et la consommation d'énergie. Par conséquent, la relation étroite entre l'interconnexion, l'algorithme de routage, et de l'architecture de routeur nécessite designers à trouver des compromis dans les paramètres de conception pour s'adapter à une application.

Nous croyons que le routage et la dégradation de la performance sont les problèmes les plus difficiles de Réseau sur puce 3D partiellement connectés verticalement. Route devrait être abordée au niveau du système. Une impasse sans mécanisme de routage efficace est nécessaire pour déterminer les chemins les paquets doivent suivre, en particulier entre les nœuds qui n'ont pas de liens verticaux. Contrairement à l'acheminement, la dégradation de la performance doit être contrôlée à la fois le système (topologie) et de l'architecture (routeur) les niveaux de conception, depuis le temps de latence de transport de paquets est la somme

des deux lien et routeur latence. Par conséquent, la performance peut être maintenue à un niveau souhaitable des liens verticaux seulement bien distribués et les routeurs 2D et 3D à faible latence.

Problème 1: Routage dans de Réseau sur puce 3D partiellement connectés verticalement

Dans entièrement connecté NoC 3D, tous les routeurs dans toutes les couches sont liées à des couches supérieures ou inférieures. En d'autres termes, tous les routeurs sont des routeurs 3D. Par conséquent, les routeurs peuvent utiliser, par exemple, une dimension d'ordre algorithme de routage sans verrouillage comme Z-Tout d'abord, quand ils ont besoin d'envoyer un paquet à une couche supérieure ou inférieure.

Au contraire, dans le sens Réseau sur puce 3D partiellement connectés verticalement, certains routeurs (nœuds) sont des routeurs 2D et ils n'ont pas de liens verticaux (TSV). Par conséquent, une question peut être soulevée. Quel chemin le paquet envoyé par les nœuds 2D (carrés pointillés) devrait suivre, quand ils ont besoin de communiquer avec un autre nœud qui se trouve dans le niveau supérieur ou inférieur? Un autre problème est interblocage. Les chemins proposés ne doivent pas créer l'impasse à travers le réseau. Par conséquent, dans une impasse et l'algorithme direct verrouillage routage libre est nécessaire.

Problème 2: la dégradation des performances dans de Réseau sur puce 3D partiellement connectés verticalement

La deuxième préoccupation principale de Réseau sur puce 3D partiellement connectés verticalement est la diminution de la performance. Manque de quelques TSV dans Réseau sur puce 3D partiellement connectés verticalement signifie moins de canaux ou, en d'autres termes, moins de bande passante. Ce problème devient encore pire pour les applications sensibles à la latence. La latence de transport de paquets, qui peut être considérée comme un indicateur de performance, est la somme des temps de latence de routeur (L_{ri}), qui dépend de la microarchitecture (voir Equ 1).

$$\text{Transport Latency} = \sum_{i=1}^n (L_{ri}) \quad \text{Equ 1}$$

Donc, le nombre et le placement des liens verticaux influencent de manière significative la trémie nombre moyen de topologie qui se termine enfin en effet significatif sur la latence moyenne, comme un indicateur de performance aussi.

Par conséquent, du point de vue du système, le "nombre", "placement", et "affectation" de liens verticaux affectent les performances de manière significative, et du point de l'architecture micro de vue de l'efficacité de routeur influences de conception performances. Alors, si ces questions sont bien résolus sur la base du modèle de trafic et les limites de conception, non seulement la performance peut être sauvé, mais aussi l'architecture finale sera plus rentable en raison de liens moins verticales (TSV) qui conduit à une plus grande surface en couches intra et un meilleur rendement.

Analyse du niveau microarchitecture

Outre les liens et leur structure d'interconnexion, de l'architecture du routeur joue un rôle clé dans la latence du réseau. La complexité du routeur est une épée à double tranchant. D'une part, les modules de contrôle complexes peuvent augmenter la fiabilité, éviter les conflits, et etc, qui est très utile et parfois nécessaire, et d'autre part, ils peuvent facilement augmenter le transport de paquets de latence en ajoutant quelques cycles supplémentaires pour plus de calculs ou envoyer et recevoir des paquets de contrôle. En outre, il peut forcer le routeur d'exiger plus d'espace et de puissance. Par conséquent, de trouver un compromis entre les avantages de l'architecture d'un routeur et son coût est un sujet complexe qui doit être prise en compte. Dans le contexte de Réseau sur puce 3D partiellement connectés verticalement, non seulement la conception du routeur devrait être aussi simple que possible et ajoute une charge supplémentaire le moins possible, mais aussi il doit gérer le mécanisme de routage basé sur l'algorithme de routage efficace.

Analyse du niveau système

Fondamentalement, le nombre minimum de liens verticaux (TSV) est déterminée en fonction de différents facteurs tels que le rendement acceptable, les exigences des applications, et etc D'autre part, dans le sens de Réseau sur puce 3D partiellement connectés verticalement, par différents lieux de liens verticaux (placement) et/ou différentes missions de 2D routeurs (nœuds) les liens verticaux (affectation) une nouvelle interconnexion (Topologie) est faite. Par conséquent, si nous considérons la topologie de chaque couche à base de maille-est, le nombre de topologies possibles (T) pour un Réseau sur puce 3D partiellement connectés verticalement est calculé comme suit:

$$T = (P \times A)^Z \quad \text{Equ 2}$$

Où P est le nombre de placements possibles, A est le nombre d'affectations possibles, et Z est le nombre de couches. De toute évidence, si le nombre de placements et les affectations d'augmenter, le nombre de topologies possibles sera augmenté aussi. Si l'on suppose que la taille de la couche, qui est le nombre de nœuds ($N = X \times Y$) dans chaque couche, et le nombre de liaisons verticales (VL) dans chacune des couches sont les mêmes dans toutes les couches, le nombre de placements possibles (P) et des missions (A) dans chaque couche peuvent être calculés comme suit:

$$P = \binom{N}{VL} = \frac{N!}{VL! \times (N-VL)!} \quad \text{Equ 3}$$

$$A = VL^{(N-VL)} \quad \text{Equ 4}$$

Le nombre de possibilités sont énormes, même pour ($N = 9$), ce qui rend l'analyse du comportement des topologies difficile et en quelque sorte impossible. Ce problème s'aggrave par une augmentation de la taille des couches (N) et la réduction de nombre de liaisons verticales (VL). Bien que par l'empilement de plusieurs couches en moyenne nombre de sauts est réduit et l'intégration devient plus dense, la question de la dissipation de la chaleur s'aggrave qui conduisent à une dégradation des performances et de fiabilité. Par conséquent,

nous croyons que la taille de dimension en couches sera plus grande que le nombre de couches dans 3D NoC.

Sur la base de cette analyse, nous pouvons voir que l'exploration architecturale de Réseau sur puce 3D partiellement connectés verticalement est à la fois critique et pas triviale. Placement et affectation des liaisons verticales à travers les couches et l'algorithme de routage utilisé une incidence directe sur la répartition du trafic en 3D NoC. Si elle est faite correctement, l'encombrement est réduit, des points chauds peuvent être évités, et enfin le seuil de saturation du réseau sera augmenté. En outre, une bonne affectation des liens verticaux à nœuds 2D peut finir en assez usage de liens verticaux et des tampons dans les routeurs 3D, ce qui signifie moins de blocages dans le réseau.

Par conséquent, la question posée ici est : comment le créateur placer correctement et affecter les liens verticaux pour empêcher ou au moins de limiter la dégradation de la performance? Par ailleurs, y a-t-il des paramètres qui par leur comportement de Réseau sur puce 3D partiellement connectés verticalement peuvent être affichées ou mesurées? Si il ya, comment peuvent-ils montrent les changements dans la performance et pourquoi? Néanmoins, le grand nombre de possibilités de topologie est un problème grave qui peut facilement menace l'évaluation de Réseau sur puce 3D partiellement connectés verticalement. Par conséquent, en proposant une méthode simple et pratique qui estime rapidement le seuil de saturation de la latence moyenne comme un indicateur de performance serait souhaitable.

CHAPITRE TROIS: ETAT DE L'ART

Akbari et al. [52] traitent le problème de faible taux de rendement en 3D NoC, en raison du processus de fabrication non-maturité de connexions TSV en proposant un nouvel algorithme de routage sans verrouillage, appelé AFRA, pour 3D NoC sur la base maille, qui peuvent tolérer lien vertical défauts. L'architecture 3D NoC concerné est couches inter et intra entièrement connectés. Quand un nœud a réalisé que son maillon vertical est défectueux, il doit envoyer son paquet à un nœud appelé "nœud de fuite" qui est trouvé par l'algorithme de calcul d'itinéraire proposé. L'algorithme de calcul d'itinéraire proposé a été prouvé d'être libre impasse. AFRA est évaluée, bien que la simulation de réseau avec une précision sur le cycle, et est comparée à acheminement adaptatif plane. Les résultats révèlent que l'AFRA améliore routage adaptatif plane dans les modèles de trafic à la fois synthétiques et réelles.

Ying et al. [54] présentent deux algorithmes de routage généralisées pour de Réseau sur puce 3D partiellement connectés verticalement, qui peuvent maintenir les performances du NoC et améliorer l'utilisation des TSV. La contribution principale de cet article est un algorithme de routage appelé Source basée court Manhattan et Destinations Basé court Manhattan. L'algorithme trouve les plus proches possibles TSV, à la fois près de la source et de destination nœuds de source et de destination couches. Ensuite, l'algorithme proposé détermine dans chaque couche intermédiaire du chemin d'accès approprié basé sur la position des TSVs calculées. Les expériences ont été réalisées en SystemC et RTL. Ils comparent les résultats de différents paramètres tels que le temps d'exécution, débit moyen, système d'interconnexion et TSV consommation d'énergie, et l'utilisation TSV avec le rendement de 100% (entièrement connecté), 50%, et 25% canal densité verticale. Enfin, ils ont montré que les configurations de densité des canaux verticaux 50% sont équilibre raisonnable pour être la meilleure solution de conception, car ils peuvent maintenir les performances du système par

rapport à la densité de canal vertical de 100% et Z- première routage tandis que son ratio de rendement est de 0,9.

Park et al. [63] proposent une 3D empilés NoC architecture d'un routeur, appelé MIRA, qui, contrairement aux routeurs 3D dans des travaux précédents, est empilée en plusieurs couches et optimisé pour réduire les exigences en matière de superficie totale et la consommation d'énergie. Ils discutent des détails de conception 3D de routeur multi-couches (3DM) et sa version améliorée avec chaînes rapides supplémentaires appelé 3D routeur multi-couches avec des chemins express (3DM-E). Enfin, ils les comparent contre une (6×6) conception 2D et un routeur de base de 3D de conception. Toutes les conceptions sont évaluées en utilisant un simulateur 3D NoC précis du cycle, et intégré avec le modèle de puissance d' Orion pour l'exécution et l'analyse de puissance. Les résultats de la simulation avec des traces synthétiques et d'application démontrent que les routeurs NoC multi-couches proposées (la 3DM et 3DM-E) peuvent surpasser la 2D et des modèles 3D naïfs en termes de performance et de puissance. Il peut atteindre jusqu'à 42% de réduction de la consommation d'énergie et l'amélioration jusqu'à 51 % de temps d'attente moyen des charges de travail de synthèse. Avec des charges de travail réelles, ces avantages sont autour de 67% et 38%, respectivement.

Dans Bartzas et al. [55] (élargi dans le premier chapitre de Gebali et al. [56]), aborder la question de la réduction du nombre de liens verticaux dans 3D NoC à base de maille suivant des schémas spécifiques pour adapter la topologie des communications de l'application des flux par une pleine méthodologie d'exploration logiciel pris en charge. Leur méthodologie d'exploration est en mesure d'évaluer les topologies 3D basé modèle et proposer celles qui répondent aux contraintes de conception meilleure. Malheureusement, ils n'expliquent pas en détail l'algorithme de routage basé sur l'utilisation des destinations temporaires dans les couches intermédiaires. En outre, chaque routeur dispos d'une table de routage qui repose sur les adresses de source/destination décide le port de sortie de chaque paquet entrant. Ils évaluent l'employant d'exploration, l'extension du simulateur Worm - Sim NoC, et le nourrir avec différents types de trafic (uniformes, transposer, et hot spot). Leur fonction principale de coût est la consommation d'énergie, avec les autres facteurs de coûts sont le temps de latence moyen de paquets et la surface totale du bloc de commutation. La base de comparaison sont les NoC entièrement interconnectés verticalement à base de maille 3D de 64 nœuds (4×4× 4) et de 144 nœuds (6×6×4) et les NoC de 64 nœuds à base de maille 2D (8×8) et de 144 nœuds (12×12). Enfin, ils montrent que les NoC sur la base-Mesh 2D ont plus la consommation d'énergie, une plus grande latence moyenne d'un paquet, et moins de consommation de la surface totale du bloc de commutation que verticalement entièrement connecté 3D NoC.

Un autre travail le plus lié à cette thèse, est Xu et al. [59]. En raison de la criticité de trouver un équilibre entre la performance et le coût de fabrication en 3D NoC, Xu et al. [59] modéliser un 3D NoC, étudier la mise en place de TSV et le coût de TSV, et d'analyser la performance avec un nombre différent de TSV à l'aide d'un simulateur de système complet. Comme une étude de cas pour trouver le placement optimal des TSV, ils ont choisi une couche de maille 8×8. Le nombre d'emplacements possibles des liaisons verticales 16 dans une maille de 8×8 est trop grande. Par conséquent, ils explorent les possibilités à partir d'un

NoC plus petit en écrivant un programme qui énumérer de manière exhaustive toutes les possibilités de placement et de la production des combinaisons de plus faible moyenne Hop-Count. Ils utilisent la méthode de diviser pour régner pour traiter le grand espace d'exploration. Ils divisent une couche de 8×8 Mesh avec 16 liens verticaux en quatre 4×4 couches de maille, chacune avec 4 liens verticaux. Enfin, en analysant les résultats des placements possibles (par simulation exhaustive) dans une couche 4×4 Mesh ils ont trouvé deux modèles dont deux sont tels que les liens verticaux sont exactement trois sauts loin de l'autre et dans chaque ligne et colonne il ya un vertical lien. Ils essaient de placer les motifs sur une couche de maille 8×8 avec différentes positions pour trouver les optimal liens verticaux placement. Leurs résultats expérimentaux ont montré que la comparaison avec 8 liens verticaux (25 % de nœuds sont des routeurs 2D) à 8×8 couche de maille, les latences du réseau moyenne de 8×8 couche Mesh avec 64 liens verticaux (vertical entièrement connecté) et 16 verticale liens (50 % de nœuds sont des routeurs 2D) sont réduits respectivement de 14.7% et 7.38%.

Ils supposent que le nombre et l'emplacement des TSV sont les mêmes dans chaque couche. En outre, ils proposent un schéma régulier fixé pour la mise en place des TSV dans lequel un nœud 3D est fixé à trois nœuds 2D seulement par un nombre de sauts. Cela pourrait être un goulot d'étranglement sur la base des opérations des nœuds attachés. Aux NoC de tailles irrégulières et le nombre irrégulier de liens verticaux, par exemple une maille de 13×13 avec 53 piliers, ne sont pas adaptées à leur méthode qui est une limitation considérable. Cependant, leur idée peut être étendue aux NoC avec des tailles régulières et le nombre régulier de piliers, par exemple 8×8 maille avec 4 liens verticaux (répartis en 4 mailles, chaque 4×4 avec 1 lien vertical), ou une grille 9×9 avec 18 liens verticaux (répartis en 9 mailles, chaque 3×3 avec 2 liens verticaux).

Chen et al. [60] évaluer aussi un NoC à base de FPGA 3D verticalement partiellement connecté. Ils introduisent deux méthodes pour réduire la consommation de la zone : a) moins le nombre de TSV dans chaque routeur 3D et b) moins le nombre de routeurs 3D à travers chaque couche. En outre, ils évaluent une architecture hybride sur la base de ces deux méthodes. Cependant, ils supposent que 1) tous les étages ont la même dimension, 2) tous les routeurs 3D, dans le premier procédé, doivent avoir le même nombre de TSV dans toutes les couches, et 3) le nombre de routeurs 3D, dans la deuxième procédé, doit être le même dans toutes les couches. Par conséquent, bien que ce travail soit intéressant, plus de souplesse est nécessaire pour répondre aux besoins d'une évaluation complète des différentes possibilités et les comportements d'architecture. Enfin, les auteurs recommandent plusieurs configurations pour les architectures génériques 3D FPGA, ce qui permet d'économiser jusqu'à 52% de la surface.

CHAPITRE QUATRE:

ALGORITHM DE ROUTAGE ELEVATOR FIRST

Dans de Réseau sur puce 3D partiellement connectés verticalement, si les nœuds source et destination d'un paquet sont dans des couches différentes, mécanisme de routage algorithmique dimension ordonnée, comme Z - Tout d'abord, ne peut pas être utilisé comme le réseau est potentiellement irrégulier. Le problème se produit quand un nœud, qui ne possède pas de liens verticaux, veut envoyer un paquet à la couche supérieure ou inférieure. Dans ce cas le nœud peut utiliser un nœud intermédiaire dans la même couche dite ascenseur (route 3D) qui est relié à UP et/ou orifice(s) vers le bas. Par conséquent, l'idée principale derrière Elevator First algorithme de routage est l'utilisation de nœuds intermédiaires (Ascenseurs) qui ont des liens verticaux pour envoyer les paquets de nœuds qui n'ont pas les. Ascenseur dispose de trois types, puisque dans sur puce 3D partiellement connectés verticalement certains routeurs (nœuds) n'ont pas de liens verticaux (ports HAUT ou/et BAS). Si un routeur ne dispose que d'un port UP et est choisi pour transmettre les paquets de couches supérieures est appelé-Ascenseur (Elv_u). En outre, si un routeur ne dispose que d'un port DOWN et est choisi pour transmettre les paquets de couches inférieures est appelé BAS-Ascenseur (Elv_d). Si un routeur a deux ports de sortie HAUT et BAS et est choisi pour transmettre les paquets de couches supérieures et inférieures, il est appelé un UP/DOWN-Ascenseur (Elv_u j).

Dans cette section, la procédure de prise de décision et le routage des paquets en Elevator First algorithme de routage sont présentés. Pour utiliser Elevator First algorithme de routage, chaque nœud de Réseau sur puce 3D partiellement connectés verticalement doit être attribuée à deux adresses: a) l'adresse d'un Elv_u ou Elv_u/d et b) l'adresse d'un Elv_d ou Elv_u/d. Par conséquent, lorsqu'un routeur (nœud) a un paquet qui doit être transmis à une couche

supérieure ou inférieure, ce routeur transmet directement le paquet à sa ascenseur UP ou DOWN ascenseur.

Flag_T: Si T drapeau est à '1', cela signifie que la tête est un en-tête courant temporaire. Tandis que si le drapeau T est '0', cela signifie que la tête est un courant permanent et de l'en-tête du paquet principal.

Flag_U: Si l'indicateur U est '1', cela signifie que le paquet doit être transmis à la couche supérieure. Alors que si l'indicateur U est '0', cela signifie que le paquet doit être transmis à la couche inférieure.

Lorsque l'ascenseur reçoit l'en-tête temporaire en cochant la T drapeau, il se rend compte que cet en-tête est temporaire et en vérifiant le drapeau U il se rend compte que la destination réelle est dans la couche supérieure. L'en-tête provisoire est ensuite enlevé et le paquet simplement transmis au port de sortie UP.

Contrairement à Z-Premier algorithme de routage comme un algorithme de commande de dimension est gratuit impasse, à l'origine Elevator First algorithme de routage n'est pas une impasse sans. Depuis le routage est vortex avec une capacité limitée de tampons, impasse qui se passe. Pour faire face à problème de blocage dans Elevator First algorithme de routage, les canaux du plan 2D doivent être en mesure de séparer les paquets où les destinations sont dans les couches supérieures des paquets où les destinations sont dans les couches inférieures. Nous avons donc introducteur à channes virtuels pour chaque canal physique dans le plan 2D et de les appeler Z^+ et Z^- . Le réseau virtuel Z^+ est composée de $2D + Z$ ports de canaux virtuels et des canaux verticaux ascendants qui contiennent le port de sortie HAUT et BAS port d'entrée. De même, le réseau de Z^- virtuel contient le port d'entrée UP et le port de sortie DOWN. Le Z^+ et Z^- réseaux virtuels sont entièrement indépendantes l'une de l'autre et les paquets d'un réseau virtuel ne passent à travers l'autre. La preuve formelle d'Elevator First liberté de routage de blocage est présentée dans [68].

Fondamentalement, dans une architecture basée sur un treillis, les adresses se composent d'un triplet (X,Y,Z). Elevator First modules de routage traite de quatre types de discours à décider d'un port de sortie appropriée : a) actuelle adresse du routeur (ADRC), b) Adresse de paquets de destination (ADRd), c) Adresse Elv_u (ADRez+), et d) Elv_d Adresse (ADRez-). Les adresses des deux Elv_u et Elv_d sont initialement affectés à chaque nœud et sont enregistrées dans deux registres. Notez que même si un routeur a une place ou/et un port de sortie DOWN, les adresses de son Elv_u et Elv_d peuvent être différents de sa propre adresse, par exemple, en raison de la répartition du trafic que le concepteur peut désirer.

Chaque accès d'entrée comporte un module de calcul d'itinéraire qui effectue Elevator First algorithme de calcul d'itinéraire. Tout d'abord, lorsque le module de routage reçoit un en-tête, il compare le Z partie de l'adresse de destination (ADRd) du paquet, Z_d , avec Z partie de l'adresse du routeur (ADRC), Z_c . Si Z_d est le même que Z_c (c'est à dire les nœuds source et de destination sont dans la même couche), une machine à états finis round-robin (RR FSM) qui décide des réseaux de Z^+ ou Z^- virtuel doit être sélectionné. La stratégie round-

robin, par qui, si le paquet précédent a été transmis par exemple à la Z+ réseau paquet courant devrait être transmis à la Z- réseau, vise à équilibrer la charge entre les deux réseaux virtuels.

D'autre part si Z_d n'est pas la même que plane Z_c , un module de calcul d'itinéraire LOCAL compare ADRC avec les adresses correspondantes ascenseur (ADRez+ ou ADRez-). Si le nœud courant est son propre ascenseur, ce qui signifie que le routeur de courant a le lien vertical souhaité, le port de sortie 3D correspondant sera choisi pour transmettre le paquet. Dans le cas contraire, le port d'entrée LOCAL ajoute un en-tête de paquet temporaire avec Elv_u ou Elv_d adresse (en ce qui concerne la couche de destination) en tant que destination temporaire et intermédiaire et transmet le paquet au port de sortie en 2D correspondant (déterminée par X- premier algorithme de calcul d'itinéraire) de le réseau virtuel correspondant (Z+ ou Z-).

Tandis que l'arbitrage entre les paquets simultanés ciblant une même sortie est fait dans les ports de sortie, toutes les décisions de routage sont expédiées dans les ports d'entrée pour envoyer des paquets vers le port de sortie approprié. Ajout têtes temporaires se fait dans les ports d'entrée 3D comme au port d'entrée LOCAL. De même, la suppression des en-têtes temporaires est effectuée dans des ports d'entrée 2D lorsque le paquet doit aller vers le haut ou vers le bas. Lorsque les ports d'entrée 2D reçoivent un paquet, d'abord ils vont vérifier si l'en-tête de paquet est temporaire ou permanente (par le biais de drapeau T). Si l'en-tête est temporaire et le nœud en cours est la destination, cela signifie que le routeur est le courant ascenseur. Par conséquent, le port d'entrée 2D supprime l'en-tête temporaire et transfère le paquet vers les ports 3D correspondant sur la base de la valeur du drapeau de U. Dans le cas contraire, si l'en-tête est stable, cela signifie que la couche en cours est la couche de la destination et par conséquent, les 2D Utilise des ports d'entrée X-premier algorithme de calcul d'itinéraire pour trouver le port de sortie vers la destination.

Résultats expérimentaux d'Elevator First algorithme de routage

Afin d'étudier la faisabilité d'Elevator First algorithme de routage d'un modèle SystemC cycle-précis d'Elevator First routeur a été mise au point. Nous avons mesuré les temps de latence moyens des (5×5×5) réseau par rapport au débit d'injection de flit (charge offerte de noyaux, comme la fraction de la charge maximale possible, c'est à dire une flit par cycle) dans différentes situations pour l'utilisation de distribution de trafic uniforme (dans laquelle chaque noyau envoie des paquets de manière uniforme à tous les autres noyaux avec une même probabilité et un même taux) et localisée (quand destinations les plus proches d'être atteint ont une probabilité plus élevée).

Pour ces simulations routeurs avec 16-flit FIFO de profondeur et de 16 flit taille des paquets ont été utilisés. La courbe "Uninorm" présente la latence moyenne d'un réseau entièrement connecté 3D-Mesh en utilisant le Z-Premier algorithme de routage ordinaire. La courbe " Uni0 " présente la latence moyenne en utilisant Elevator First algorithme de routage lorsque 0% des liens verticaux sont supprimés. Cela signifie que le réseau est un 3D-Mesh entièrement connecté, mais il utilise deux canaux virtuels (tampons) de Z + et Z - pour monter et paquets descendant respectivement, et le (haut et bas) Ascenseur de chaque nœud

est lui-même. Comme on peut l'observer, la séparation du montant et descendant en deux paquets Z^+ et Z^- indépendant des réseaux a une influence positive sur la performance du réseau, comme la circulation et par conséquent la probabilité de contention est divisé par deux. Pour faire la comparaison un peu plus juste, le réseau ordinaire avec de plus grandes FIFO ont été simulées à avoir le même montant global des ressources comme "Uni0" avec 2 VC. La courbe "UniNormLF" présente ce cas et comme on peut le voir la performance du réseau est meilleure que dans le cas de "Uninorm", mais il existe encore un écart entre la performance de "Uni0" et "UniNormLF" même si les ressources de mise en mémoire tampon pour les deux cases sont les mêmes. Les courbes "Uni5", "Uni10", "Uni25", et "Uni50" sont les résultats lorsque 5%, 10%, 25% et 50% de liaisons verticales sont éliminés respectivement.

Dans notre plate-forme de simulation, le choix des liens verticaux à enlevé est choisi au hasard, et afin d'avoir une répartition plus uniforme de manquer liens verticaux les simulations ont été répétées 20 fois. Notez que nous n'avons pas imposé de contraintes sur la sélection de liens verticaux à enlever et aux topologies modélisées Il pourrait y avoir des routeurs sans lien ascendant (UP PORT), descendant lien (BAS port), ou soit. Le seul point doit être pris en considération, c'est que dans chaque couche il doit y avoir au moins un ascendant et un canal descendant afin de maintenir le réseau connecté. Pour les nœuds avec lien supprimé verticale (s), un (deux) Ascenseur(s) est (sont) choisi au hasard entre les routeurs les plus proches dans laquelle le lien vertical correspondant n'est pas supprimé.

Dans la distribution de circulation régionale la probabilité d'émission d'un paquet vers une destination décroît exponentiellement avec la distance de la destination. En d'autres termes, avec une distribution de circulation régionale, des noyaux envoient des paquets à des nœuds plus proche et inférieur à celles plus éloignées. Même lorsque 10% des liaisons verticales sont éliminés de la performance du réseau est approximativement la même que celle d'un réseau 3D-Mesh entièrement connecté en utilisant l'habituel Z-First algorithme de routage (par exemple "de LocNorm").

Elevator First algorithme de routage n'est pas minimaliste et le chemin entre la source et la destination peut être plus longue que la distance de Manhattan. Le nombre de sauts moyens qui montre le nombre moyen de routeurs qu'un paquet doit passer pour atteindre sa destination pourrait être un bon paramètre pour décrire comment Ascenseur Première routage dégrade la longueur du trajet. Le houblon compte non seulement donne une bonne approximation de latences de paquets dans les charges de faible trafic, mais aussi avoir une influence directe sur la consommation d'énergie de paquets pour arriver aux destinations. En supposant que la consommation d'énergie moyenne de chaque routeur est P et le nombre de sauts est la moyenne de n paquets, on peut estimer à peu près que la puissance consommée par le réseau est proportionnelle à $n \times P$. Bien sûr, c'est une estimation approximative et à mesurer la consommation d'énergie réelle où il ya beaucoup d'autres paramètres qui doivent être pris en considération. Nous avons mesuré la trémie comptes moyens des topologies mentionnées ci-dessus pour les deux types d'uniformes et les distributions de trafic localisées. Dans ces expériences l'un des routeurs les plus proches 3D est attribué comme l'ascenseur de routeurs, et même en changeant cette stratégie, les variations moyennes hop - count. Comme

CHAPITRE QUATRE: ALGORITHM DE ROUTAGE ELEVATOR FIRST

on le voit, en augmentant le nombre de liens verticaux supprimés les hop - count moyenne augmente. Et, bien que le nombre de sauts moyens dans le cas du trafic localisé soient moindre que dans le cas de trafic uniforme, Elevator First routage dégrade le Hop-Count moyenne de l'affaire de trafic localisée proportionnellement plus que dans le cas de trafic uniforme. Cependant, même avec 50 % manquant liens verticaux dans la répartition du trafic localisée, la moyenne dégradation nombre de sauts est d'environ 18 %. Dans la plupart des situations, une telle dégradation semble être acceptable.

CHAPITRE CINQ:

ELEVATOR FIRST ROUTEUR MICROARCHITECTURE

Le routeur 2D DSPIN [73] a cinq ports de l'Ouest, Nord, Est, Sud, et local. Le routage et l'arbitrage sont effectués de manière distribuée dans chacun des ports d'entrée et de sortie respectivement. Le port d'entrée est responsable de recevoir les (flits) d'un paquet et de les stocker dans une FIFO. En outre, le module de routage au port d'entrée effectue l'algorithme de routage afin de déterminer le port de destination. D'autre part, dans le port de sortie, le module d'arbitrage décide quel port d'entrée, parmi celles qui ont déjà (simultanément) un paquet à envoyer, doit être servi. L'arbitrage choisit le port d'entrée approprié par l'émission du signal de sélection du multiplexeur de sortie. Changer de tissu est les connexions internes du routeur entre les ports.

En routeur DSPIN quand un port de sortie veut lire de la mémoire tampon situé dans le port d'entrée a) le signal Read (qui est utilisé comme la commande de lecture de la mémoire tampon) doit être activée ; b) le signal ROK (pour "Lire OK" qui représente ce que le tampon n'est pas vide, alors il est une donnée valide à transférer) doit être activé. Identique, quand un port de sortie veut écrire dans le tampon d'entrée du routeur en aval , a) le signal d'écriture (qui est utilisé comme commande d'écriture de la mémoire tampon) doit être activé , b) le signal de WOK (pour « Écrire OK " qui représente ce que le tampon en aval du routeur n'est pas plein) doit être activé . Enfin, si le routeur en aval a suffisamment d'espace dans sa FIFO d'entrée, le signal de WOK devient permettant, ce qui permet aux flits dans le port d'entrée ' i ' à écrire dans la FIFO en aval du port d'entrée. Techniquement, les flits sont écrits lorsque les deux signaux "WOK" et "ROK" sont '1 ' qui signifie qu'il ya un flit dans le port d'entrée ' i ' et il ya suffisamment d'espace dans le port d'entrée en aval.

En outre, dans DSPIN NoC dans chaque flit d'un paquet, il ya deux bits réservés (drapeaux) a appelé "la balance des paiements", qui indique de commencer paquet, et " EOP", qui indique la fin de paquet. Le flit qui drapeau "BOP" est '1 ' est la tête et contient l'adresse de destination du paquet. Aussi un flit qui " EOP" est '1 ' est la remorque du paquet. En outre, dans une entrée (input 'i') port lorsqu'un flit d'un paquet arrive dans la FIFO, "ROK" devient '1' ce qui signifie que l'entrée FIFO n'est pas vide. Par conséquent, si les deux signaux "BOP" et "ROK" sont '1', les modules de routage effectue le routage pour indiquer le port de sortie du paquet (sortie 'j') en envoyant une demande "Req_i_j" à l'arbitrage de sortie. Le module d'arbitrage dans le port de sortie sélectionne une demande parmi ceux reçus sur la base pour la ronde, et permet le signal prêt "Rdy_j_i" qui est l'entrée de sélection de 'j ' multiplexeurs la sortie pour montrer sa volonté de recevoir les flits du paquet. De cette manière, l'orifice de sortie 'j' est affecté à l'orifice d'entrée «i» jusqu'à ce que la fin du paquet.

L'Elevator First routeur est composé de trois types de ports : 1) les ports 2D (Nord, Ouest, Sud, Est), 2) ports 3D (haut et bas), et 3) le port local et deux tissus Z+ et Z- commutateur. La microarchitecture d'Elevator First routeur est basée sur les caractéristiques de Réseau sur puce 3D partiellement connectés verticalement et Elevator First algorithme de routage. Comme décrit précédemment dans le chapitre quatre, pour éviter les blocages, les paquets qui vont à couches supérieures utilisent le réseau virtuel de Z+, et les paquets qui vont vers les couches inférieures utilisent le réseau Z- virtuel. Du point de vue du routeur, le port d'entrée de BAS qui reçoit les paquets de couche inférieure et l'orifice de sortie UP qui envoie les paquets à la couche supérieure appartenant à Z+ réseau virtuel. De même, l'orifice d'entrée UP qui reçoit les paquets de la couche supérieure, et le port de sortie DOWN, qui envoie les paquets à la couche inférieure appartiennent au réseau virtuel Z- . Chacun des orifices 2D comporte pratiquement deux Z+ et Z- ports indépendants qui peuvent transmettre les paquets appartiennent à savoir si Z+ ou Z- réseau virtuel. Pour résumer, nous appelons Z+ réseau virtuel, VN+, et le réseau Z- virtuel, VN-. Comme les paquets de chaque réseau virtuel ne devraient pas utiliser l'autre réseau virtuel pour atteindre leur destination, nous définissons deux commutateurs différents tissus Z+ et Z- (SF+ et SF-) dans l'Elevator First routeur de séparer complètement la Z+ et Z- trafics.

Non seulement il n'existe aucun lien entre le Z+ et Z- Switch Fabric, mais aussi il n'ya pas de connexion entre les ports nord/sud d'entrée et des ports de sortie Est/Ouest, en raison du fait que le libre X-Premier algorithme de routage afin de dimension impasse a été utilisé pour le routage intra paquet de couche. En effet, les paquets sont acheminés tout d'abord dans la direction X pour être alignée avec la partie Y de leur destination, et ensuite, dans la direction Y pour atteindre la destination.

Ajout et suppression de temporaire tête en Elevator First routeur

En de Réseau sur puce 3D partiellement connectés verticalement, quand un nœud 2D veut communiquer avec un autre nœud dans une autre couche, l'Elevator First algorithme de routage transmet les paquets du nœud 2D à son assigné ascenseur en ajoutant un en-tête temporaire qui contient l'adresse de l'ascenseur. Lorsque les paquets arrivent à l'ascenseur, l'en-tête temporaire doit être enlevé et le paquet est transmis à un port 3D correspondant.

D'un point de vue microarchitecture, l'ajout d'un en-tête temporaire est toujours fait dans les ports d'entrée LOCAL et 3D et la suppression d'un en-tête temporaire se fait toujours dans le port d'entrée 2D. Etant donné que chaque port d'entrée 2D a deux séparé (Z+) et (Z-) des réseaux virtuels, (2D Z+) port d'entrée enlève les en-têtes temporaires de (Z+) des paquets de réseau virtuel et (2D Z-) port d'entrée enlève les en-têtes temporaires de l'(Z-) les paquets de réseau virtuel. En outre, comme le port d'entrée UP appartient à (Z-) réseau virtuel, il ajoute le (Z-) tête temporaire (Z-) des paquets de réseau virtuel et que le port d'entrée de BAS appartient à (Z+) de réseau virtuel, il ajoute l' (Z+) de la tête temporaire (Z+) paquets de réseau virtuel. Contrairement 2D et 3D ports d'entrée, port local n'appartient à aucun (Z+) ni (Z-) réseau virtuel. Comme port local est totalement indépendant, plus des deux (Z+) et (Z-) tête temporaire est possible dans le port d'entrée LOCAL. Il est déterminé sur la base de la couche de destination du paquet de position par rapport à la position de la couche de source de paquet.

La microarchitecture des ports d'Elevator First routeur

Dans cette section, la microarchitecture des ports Elevator First routeur est expliquée en détail. Depuis le port local est la porte d'entrée des paquets à la verticale-partiellement-Connecté 3D-NOC, il est d'abord expliqué. Port local est le port le plus compliqué dans Ascenseur-Première architecture d'un routeur. Ceci est dû à deux raisons. La première raison est que le port d'entrée LOCAL est le seul port qui devrait décider de réseau virtuel du paquet. La deuxième raison est la logique d'arbitrage dans le port de sortie LOCAL. Le module d'arbitrage ne doit pas seulement décider du prochain port d'entrée pour être servi sur la base de l'état précédent, mais aussi doit indiquer lequel des réseaux virtuels (Z+ ou Z-) doivent être servis.

Le port d'entrée LOCAL peut envoyer le paquet soit (Z+) ou (Z-) réseau virtuel basé sur l'adresse de destination du paquet. Pour ce faire, le port local ne contient qu'un (Elevator First) module de l'algorithme de routage, une FIFO, et un (Elevator First) la machine à états finis pour décider du type de canal virtuel le paquet doit aller dans et ajouter un en-tête temporaire. Le port de sortie est symétrique semblable au port d'entrée. Par conséquent, à la fois (Z+) et (Z-) données et des signaux de contrôle sont reliés au port de sortie LOCAL qui rend le module et les multiplexeurs d'arbitrage LOCAL plus complexe que les ports de sortie des autres ports. LOCAL (Elevator First) FSM dans le port d'entrée LOCAL permet non seulement le signal de commande de Sel_AH à manipuler écriture et de lecture des signaux lors de l'ajout d'une nouvelle tête, mais aussi choisit le réseau virtuel approprié pour un nouveau paquet d'arrivée avec le signal VN_Sel. Lorsque Sel_AH est activé, le tampon du port d'entrée ne peut pas recevoir le signal de demande de lecture, si l'en-tête permanent du paquet ne sera pas supprimé. En même temps, Sel_AH permet au signal de demande d'écriture du routeur suivant, de sorte que l'en-tête temporaire sera écrit au lieu de l'en-tête permanent. Sel_AH doit être désactivé lorsque le buffer du port d'entrée suivant reçoit correctement les données, c'est à dire juste dans le cycle suivant après quand WOK (Lire) est de permettre. Dans les prochains cycles d'horloge permanent et l'en-tête du reste des flits jusqu'à ce que la fin du paquet sont transférés au port de sortie en 2D.

Fondamentalement, le LOCAL (Elevator First) machine à états finis est composé de deux touches HAUT et BAS (Elevator First) machines à états finis. Comme port d'entrée LOCAL, la procédure d'addition est la même dans les ports d'entrée 3D. Un point qui doit être mentionné est que le port d'entrée UP ajoute down-Ascenseur-tête temporaire pour les paquets de (Z-) du réseau virtuel et l'entrée BAS orifice ajoute-Ascenseur-tête temporaire pour les paquets de (Z+) de réseau virtuel.

Comme les deux réseaux virtuels sont complètement séparées l'une de l'autre pour éviter une impasse, un port 2D est séparé en deux réseaux virtuels de (Z+) et (Z-). Chacun des (Z+) et (Z-) ports d'entrée contient un Elevator First module de routage, une FIFO, et un ascenseur première machine à états finis pour supprimer l'en-tête temporaire des paquets. De plus chacun de (Z+) et (Z-) ports de sortie contient un module d'arbitrage robin ronds et des multiplexeurs pour multiplexer et démultiplexer les signaux de données et de commande des ports d'entrée connectés via la matrice de commutation.

Lorsque (T) drapeau dans l'en-tête du paquet est '1', cela signifie que l'en-tête est temporaire. En outre, si l'un des accès de sortie 3D (Ru (demande de port de sortie UP) et Rd (demande de port de sortie BAS)) est sélectionné, cela signifie que le nœud courant est ascenseur et le paquet doit être supprimé. Retrait de la tête temporaire peut être fait simplement en permettant Lire pour lire le premier flit du paquet et la désactivation Ecrire pour ne pas écrire dans le port de sortie 3D (en fait dans la mémoire tampon du port d'entrée 3D du prochain routeur). (Sel_RH_Z+/Z-) sont les signaux de commande générés par (2D_FSM_Z+/Z-) machines à états finis pour supprimer l'en-tête. Ils sont activés seulement pour un cycle d'horloge. En les désactivant dans le cycle d'horloge suivant l'en-tête réel (permanent) (deuxième flit du paquet) est transféré par l'orifice de sortie 3D et le flit processus se poursuivra jusqu'à ce que le transfert de la fin du paquet.

RTL niveau de synthèse des résultats expérimentaux

Nous avons développé (Elevator First) 3D routeur en VHDL et synthétisée avec Synopsis utilisant la technologie CMOS 65 nm de STMicroelectronics (GP65LVT). Nous synthétisons le routeur avec la contrainte 0ns pour atteindre la fréquence de fonctionnement maximale. Nous mesurons la région, la fréquence d'horloge maximale, et la consommation d'énergie d'un routeur 3D (Elevator First) avec FIFO profondeur de 8 flits de 34 bits (32 bits de données et 2 bits de contrôle). Dans ce tableau routeur 5-Port représente le routeur 2D sans un port vertical, 6 ports représente un routeur 3D avec port vertical vers le haut ou vers le bas, et 7-Port représente un routeur 3D avec deux ports verticaux haut et bas. Comme dernier point, afin de comprendre la zone de tête de (Elevator First) routeur Nous comparons la zone de silicium ainsi que la zone d'un routeur 3D ordinaire avec 7 ports, et qui exploite FIFO plus grands (dont la capacité tampon est à peu près équivalent à que de (Elevator First) routeur, UniNormLF ou LocNormLF. Le (Elevator First) zone de routeur est d'environ 32% supérieur à un routeur entièrement connecté ordinaire, mais le surcoût en surface, par rapport à un routeur équivalent en termes de capacité de la mémoire tampon, n'est que de 8%.

CHAPITRE SIX:

EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTÉS VERTICALEMENT

Dans le "Chapitre problème de la définition de", nous avons expliqué le problème de la dégradation des performances dans Réseau sur puce 3D partiellement connectés verticalement en raison des liens de couches inter supprimé (liens verticaux). Aussi, nous avons décrit que ce problème peut être prise en compte à la fois de microarchitecture (routeur de mise en œuvre du matériel) et système (de la conception de la topologie) niveaux. Nous avons déjà démontré la microarchitecture de (Elevator First) routeur dans le chapitre cinq. L'exploration architecturale de Réseau sur puce 3D partiellement connectés verticalement est discutée dans ce chapitre.

Comme nous l'avons montré dans le «Chapitre problème de définition», le nombre de topologies possibles le Réseau sur puce 3D partiellement connectés verticalement est énorme. En outre, de nombreux paramètres influents sur la performance de Réseau sur puce 3D partiellement connectés verticalement comme moyen nombre de sauts, le nombre de nœuds les ascenseurs desservent, et etc. Ces questions font l'analyse des architectures difficile. Dans ce chapitre, nous allons

1. définir et étudier les paramètres qui décrivent la performance de Réseau sur puce 3D partiellement connectés verticalement
2. discuter et d'évaluer une méthode de placement et d'affectation qui garantit possible nombre de sauts minimum entre les nœuds en 2D et leurs Ascenseurs, distribue uniformément la Ascenseurs dans les couches, et essaie d'attribuer aussi les nœuds 2D pour les ascenseurs, et enfin
3. présenter une méthode quadratique basée estimation pour estimer le seuil de saturation de le Réseau sur puce 3D partiellement connectés verticalement.

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

Chaque nœud dans le Réseau sur puce 3D partiellement connectés verticalement est affecté à un UP-Ascenseur (Elv_u) et DOWN-Ascenseur (Elv_d) qui sont utilisés pour transmettre ses paquets de couches supérieures et les couches inférieures respectivement. Nous définissons une «région» comme un ensemble d'un "ascenseur" (HAUT ou BAS) et ses «nœuds 2D assignés». Par conséquent, dans chaque couche, le nombre d'ascenseurs (E) et les régions (R) sont égaux. Comme il ya deux principaux types d'ascenseurs (haut et bas), il existe deux types de régions ainsi: jusqu'à-Région (ur) et bas-Région (dr). Basé sur le concept régions, quatre principaux paramètres peuvent être définis pour montrer la performance et la structure d'une région (r_i):

- Région Degré (RD):

Région degré est le nombre de nœuds dans une région qui sont servis par la Région ascenseur. Notez que l'ascenseur lui-même est un routeur qui est connecté à un noyau IP, il est donc également partie de la région.

- Moyenne région Hop-Count:

Nombre de sauts affecte la performance (latence moyenne) et la consommation d'énergie de NoC. Dans la région (r_i), Moyenne Région Hop-Count est la moyenne des distances entre chaque membre de Manhattan (m_{ij}) et la Région ascenseur (E_i).

- Région de charge:

La charge d'une liaison est liée à la fois à l'algorithme de calcul d'itinéraire et de modèle de trafic. Nous calculons les charges de tous les liens fondés sur la répartition uniforme, sous (Elevator First) algorithme de routage. Pour chaque région (r_i), la Région-charge est la somme des charges de tous les liens de distances de Manhattan entre chaque membre de la Région et la Région ascenseur.

- Totale Région Degré:

Ce paramètre est le nombre total de nœuds d'un ascenseur doit servir. Deux régions connectées dans les couches adjacentes sont reliées verticalement via les ascenseurs de Régions. Par la suite, ces paramètres sont utilisés pour étudier et évaluer la performance de Réseau sur puce 3D partiellement connectés verticalement.

Pattern-based (Placement et affectation) Algorithme dans le Réseau sur puce 3D partiellement connectés verticalement

Nombre de sauts est l'une des questions importantes dans la conception NoC. Moins hop-count signifie moins de temps d'attente et moins de consommation d'énergie qui se termine finalement dans de meilleures performances. (Elevator First) routage de Réseau sur puce 3D partiellement connectés verticalement provoque plus de nombre de sauts, ce qui est bien sûr inévitable, en raison des liens verticaux supprimés. Cependant, si dans chaque couche les ascenseurs sont placés de telle sorte qu'ils peuvent être affectés à des nœuds 2D avec un seul nombre de sauts, le nombre de sauts moyen sera aussi petit que possible. Fondamentalement,

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

il ya $(4 \times \sum_{i=1}^{HP} i)$ nœuds qui peuvent être affectés à un ascenseur avec nombre de sauts maximal (HP). Dans lequel, il ya quatre nœuds 2D possibles qui peuvent être affectés à un ascenseur avec un maximum d'un nombre de sauts. Bien sûr, le nombre de nœuds en 2D qui sont affectés à un ascenseur sont décidées par le concepteur et peuvent être plus ou moins que quatre. Si nous supposons que tous les quatre nœuds 2D possibles sont affectés à un ascenseur (Région-Degré égal cinq) avec un seul nombre de sauts, le placement possible des régions adjacentes peut être obtenue à partir du nœud de référence. Si (X_0, Y_0) est l'adresse d'ascenseur de la Région primaire:

1. Adjacent North Elevator's address: $(x_0 + (HP), y_0 + (HP + 1)) \xRightarrow{HP=1} (x_0 + 1, y_0 + 2)$
2. Adjacent West Elevator's address: $(x_0 - (HP + 1), y_0 + (HP)) \xRightarrow{HP=1} (x_0 - 2, y_0 + 1)$
3. Adjacent South Elevator's address: $(x_0 - (HP), y_0 - (HP + 1)) \xRightarrow{HP=1} (x_0 - 1, y_0 - 2)$
4. Adjacent East Elevator's address: $(x_0 + (HP + 1), y_0 - (HP)) \xRightarrow{HP=1} (x_0 + 2, y_0 - 1)$

Cette approche peut être utilisée pour placer et affecter les ascenseurs pour les nœuds et faire une Réseau sur puce 3D partiellement connectés verticalement avec des motifs similaires. Bien sûr, le placement en fonction de modèle et méthode cession algorithme que nous allons vous expliquer ici est un exemple. Le concepteur peut adapter cette approche basée sur les exigences des modes d'application et de la circulation. Par conséquent, à ce stade, nous supposons que toutes les régions ont même nombre de sauts ($HP=1$) et de la région degré ($RD=5$) pour plus de simplicité dans l'explication et l'évaluation. Le placement et l'algorithme d'affectation des régions similaires avec le motif de quatre nœuds 2D affectés à un ascenseur avec un saut peuvent être généralisés par la définition de deux matrices (briques) qui sont appelés:

- “WestEast” brick $(x_0 \pm (HP + 1), y_0 \mp (HP))$:
In (west or east) direction and their size is $[(HP + 2) \times (HP + 1)]$
- “NorthSouth” brick $(x_0 \pm (HP), y_0 \pm (HP + 1))$:
In (North or South) direction and their size is $[(HP + 1) \times (HP + 2)]$

Le placement basée sur des modèles et de l'algorithme d'affectation doit comporter deux phases principales:

- 1) La phase d'initialisation: placer le Ascenseurs et les assigner aux nœuds dans une couche de maille sur la base de la référence de l' adresse de Ascenseur dans l'Est ou de l'Ouest et les directions Nord ou Sud.
- 2) La phase post-vérification: pour vérifier si il ya des nœuds 2D qui sont affectés aux ascenseurs qui ne sont pas sur le territoire de la couche de maille.

Les deux phases de l'algorithme a été conçu pour configurer chaque nœud (X_c, Y_c) par l'adresse de ascenseur de la région du nœud. Cet algorithme a juste besoin de l'adresse de l'ascenseur de la région de référence (X_0, Y_0) et la taille de la couche $(M \times N)$. Si nous supposons que l'adresse de l'ascenseur de la région de référence est $(0,2)$, nous avons juste

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

besoin de savoir combien de briques sont nécessaires dans (direction Est) et (Nord ou Sud direction). Ceci peut être extrait par le calcul de deux différences entre l'adresse X de la référence et les adresses de courant ($X_c - X_0$), appelé Δx_{c0} , et la différence entre l'adresse Y de la référence et les adresses actuelles ($Y_c - Y_0$), appelée Δy_{c0} , ce qui peut indiquer la direction de la ligne entre le nœud de référence et le nœud actuel. Sur cette base, la pente de la ligne qui commence à partir de la référence ascenseur et se poursuit en direction de l'Est peut être calculée comme:

$$\frac{\Delta y_{c0}}{\Delta x_{c0}} = -\frac{HP}{HP+1} \quad \text{Equ 1}$$

Par conséquent, si $(\frac{\Delta y_{c0}}{\Delta x_{c0}})$ de chaque nœud (X_c, Y_c) est supérieur ou égal à $(-\frac{HP}{HP+1})$, alors nous devons aller dans des directions Est et Nord. Sinon, si $(\frac{\Delta y_{c0}}{\Delta x_{c0}})$ de chaque nœud (X_c, Y_c) est plus petit que $(-\frac{HP}{HP+1})$, alors nous devons aller en Orient, puis en direction du Sud. Aussi, lorsque le ΔX_{c0} est nulle, si la ΔY_{c0} est supérieure ou égale à zéro, alors nous devons aller dans des directions Est et Nord. Sinon, nous devons aller dans des directions Est et du Sud. Une fois terminé la direction dans laquelle nous devons aller, nous devons indiquer le nombre de briques dont nous avons besoin. Nous déterminons le nombre de briques doivent être déplacés de l'Est (E) et du Nord (N) pour atteindre la cible ascenseur en résolvant Equ 2 & 3.

$$(HP + 1)E + (HP)N = \Delta x_{c0} \quad \text{Equ 2}$$

$$-(HP)E + (HP + 1)N = \Delta y_{c0} \quad \text{Equ 3}$$

De même, en résolvant les deux équations suivantes (Equ 4 et 5) en utilisant deux variables S et E, nous pouvons obtenir le nombre de briques à parcourir dans l'Est (E) et Sud (S) à la cible directions ascenseur.

$$(HP + 1)E - (HP)S = \Delta x_{c0} \quad \text{Equ 4}$$

$$-(HP)E - (HP + 1)S = \Delta y_{c0} \quad \text{Equ 5}$$

Nous devons mentionner que la valeur absolue arrondie de E, noté $| [E] |$, et (N noté $| [N] |$ ou S noté $| [S] |$) doit être utilisé dans l'algorithme. Par exemple, si E est égal à 1.25, une brique de l'Est doit être utilisé et si E est égal 1.64, deux briques de l'est doit être utilisé. Utilisation absolue de l'arrondissement des valeurs (E et (N ou S)) conduit à l'ascenseur le plus proche possible de le nœud actuel. Par conséquent, dans la deuxième phase (post-contrôle), nous devons répondre à cette question: sont tous les ascenseurs affectés dans les limites de maille?

Si la réponse est non, alors nous devons répondre à cette question: quelle est la plus proche ascenseur aux nœuds d'exception, comme nombre de sauts est notre principale préoccupation à ce stade. Fait intéressant, en raison du fait que le choix de Ascenseurs limites ne sont qu'une hop loin des nœuds d'exception (en X ou en direction Y), leur place ne peut être que l'un des quatre cas. Il pourrait y avoir deux possibles Ascenseurs plus proches, donc, si les deux conditions seraient correctes, un seul d'entre eux doit être sélectionné. Dans de tels cas, le

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

concepteur peut décider en fonction de la valeur Région-Degré des ascenseurs qui ont été placés à l'intérieur de la couche de maille ou autre considération de conception.

Uniforme Affectation ascenseur (UEA)

Dans cette thèse, le modèle de trafic uniforme est utilisé en raison de notre approche vers les plates-formes d'application générale de but. Par conséquent, afin d'assurer une répartition de la balance des ascenseurs entre les nœuds 2D, nous proposons une méthode simple et pratique pour définir la région degrés aux valeurs plus proches de la moyenne région degré (RD_avg) qui est calculé comme Equ 6:

$$RD_{avg} = \frac{N}{E} \quad \text{Equ 6}$$

où N est le nombre de nœuds dans chaque couche et E est le nombre d'ascenseurs. Cela signifie que, dans cette méthode, tous les ascenseurs sont affectés à environ même nombre de nœuds 2D dans toutes les couches. Cela finit dans un équilibre de la circulation difficile dans les tampons d'Ascenseurs (FIFO). Dans cette méthode générale et uniforme affectation Ascenseurs », il existe deux types de région-degré que nous appelons RD_1 et RD_2. RD_1 est la valeur plafond de la moyenne région degré (RD_avg):

$$RD_1 = \left\lceil \frac{N}{E} \right\rceil \quad \text{Equ 7}$$

RD_2 est la valeur plancher de la moyenne région-Degré (RD_avg):

$$RD_2 = \left\lfloor \frac{N}{E} \right\rfloor \quad \text{Equ 8}$$

En outre, NRD_1 et NRD_2 sont le nombre de régions qui ont Région-Degré RD_1 et RD_2 respectivement:

$$NRD_1 = N - (RD_2 \times E) \quad \text{Equ 9}$$

$$NRD_2 = \frac{N - (NRD_1 \times RD_1)}{RD_2} \quad \text{Equ 10}$$

Résultats expérimentaux de méthode uniforme d'affectation ascenseur

Depuis placement a également un rôle important dans Réseau sur puce 3D partiellement connectés verticalement, nous évaluer la performance de la méthode d'attribution ascenseur uniforme basée sur deux stratégies de placement différentes :

Le règle de la première stratégie de placement:

- Les Ascenseurs peuvent être ajustés à n'importe quel endroit (adresse) s'ils se connectent à leur 2D nœuds avec le moins possible hop -count (s).

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

Les règles de la deuxième stratégie de placement:

- Placement des ascenseurs doit être sur les bords et spécialement dans les coins d'une couche de maille, tandis que leurs nœuds 2D affectés devraient être choisis à partir de la zone de point chaud au centre de la couche de maille.
- Placement des ascenseurs dans les mêmes réseaux (ex: réseau UP) dans les couches de maillage adjacentes devrait être différent.
- Placement des ascenseurs dans les différents réseaux (UP et réseaux BAS) dans une couche de maille devrait être différent.
- X-Premières chemins de routage des nœuds 2D à leurs Ascenseurs dans différents réseaux (HAUT et BAS réseaux) dans une couche de maille doivent être différents et ne se chevauchent pas, autant que possible.

Sur la base de la méthode d'assignation ascenseur uniforme où deux échantillons générés manuellement sur la base des première et seconde stratégies de placement décrites pour un nombre différent d'ascenseurs. Dans les topologies choisies, le nombre de nœuds (N) est de 25 (5×5), le nombre de couches (Z) est de 5. En outre, de comparer les résultats des uniformes topologies d'affectation ascenseur, nous utilisons les valeurs de seuil de saturation de la meilleure des topologies 18.5K avec le même (N), (E), et (Z) qui ont été au hasard (placement aléatoire et aléatoire affectation) généré.

Lorsque le nombre d'ascenseurs (E) est proche du nombre de nœuds dans chaque couche (N) et '0', les échantillons d'affectation ascenseur uniformes sur la base de la deuxième stratégie de placement surpasse par rapport à la première et la meilleure topologie de 18.5K échantillons. Lorsque (E) est proche de (N), la plupart des nœuds sont Ascenseurs et si nous avons choisi un nœud 2D de la zone hots-pot qui envoie ses paquets sur les bords ou les coins, non seulement nous réduisons la latence d'arbitrage du routeur 2D en hot-spot, mais transmet également la charge supplémentaire de ascenseur Première routage pour les domaines (périmètre et de coin) avec moins de liens et des tampons utilisation et à la Ascenseurs avec moins de nombre de ports 2D. D'autre part, lorsque (E) est proche de "0", la plupart des nœuds n'ont pas de liens verticaux et ont déjà été affecté à petit nombre de Ascenseurs. Par conséquent, peu d'ascenseurs doivent desservir un grand nombre de nœuds 2D. Dans cette situation, ce n'est pas vraiment combien de houblon les paquets de nœuds 2D doivent traverser pour atteindre leurs ascenseurs. Toutefois, le placement des Ascenseurs occupés sur les bords et les coins avec moins de trafic peut améliorer légèrement les performances.

Lorsque le nombre de Ascenseurs (E) est proche et plus ($\frac{N}{2}$), les premier et deuxième stratégies effectue presque similaire, et bien sûr, ils ont un meilleur seuil de saturation de la meilleure topologie des échantillons 18.5K. Alors que, lorsque le nombre de Ascenseurs (E) est réduite à moins de ($\frac{N}{2}$), les échantillons d'affectation ascenseur uniformes fondées sur la

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

première stratégie de placement surpasse les autres. Cela signifie que nombre de sauts joue un rôle important lors de (E) est inférieure à $(\frac{N}{2})$ et non à proximité de '0' à le Réseau sur puce 3D partiellement connectés verticalement. Dans un tel cas, le nombre d'ascenseurs est inférieur au nombre de nœuds 2D. Donc, si nous mettons tous les ascenseurs sur les bords et les coins et les nœuds 2D sur le centre puis plus de $(\frac{N}{2})$ de nœuds veulent envoyer leurs paquets pour les bords et les coins. Ce n'est pas seulement ne réduit pas l'ascenseur Première trafic dans une couche de maille, mais augmente également la charge de trafic dans la zone hot-spot. C'est pourquoi placement distribué des ascenseurs au plus près de leurs nœuds 2D assignées peut être une bonne façon d'économiser de la performance quand (E) est inférieure à $(\frac{N}{2})$ et pas près à '0'.

En outre, les seuils de saturation des échantillons d'affectation ascenseur uniformes sur la base de la première stratégie de placement sont presque similaires à partir de (E) est égale à 24 (E est égale à 13). Cela signifie que, par la réduction de 12 ascenseurs dans chaque couche (48 dans le réseau), la performance peut être sauvegardée. En outre, les seuils de saturation des échantillons uniformes d'affectation ascenseur sur la base de la deuxième stratégie de placement sont presque similaires à partir de (E) est égale à 24 pour (E) est égal à 17, qui montre, par la réduction de 8 ascenseurs dans chaque couche (32 dans le réseau) l' la performance peut être sauvé. Par conséquent, cela signifie que les uniformes des topologies ascenseur affectations peuvent tolérer plus de charges de réseau avec moins de nombre d'Ascenseurs (liens verticaux).

Enfin, il faut mentionner que, bien que ces résultats sont intéressants, en raison du grand nombre de topologies possibles, nous ne pouvons pas vraiment savoir la meilleure configuration. Cependant, nous pouvons dire qu'uniforme méthode d'affectation ascenseur peut obtenir réseau avec des seuils de saturation au-dessus de la moyenne.

Méthode d'estimation base de quadratique

Dans cette section, nous présentons une méthode d'estimation quadratique base. Nous crayons que, puis que la performance de Réseau sur puce 3D partiellement connectés verticalement est influencée par différents types de paramètres architecturaux, qui ont déjà été décrites dans la section 2, la multiplication mutuelle des paramètres dans une forme quadratique peut prédire le comportement des Réseau sur puce 3D partiellement connectés verticalement. La forme quadratique est calculée comme suit:

$$(P_1\beta_1 + P_2\beta_2 + \dots + P_n\beta_n) + (P_1P_2\beta_{n+1} + P_1P_3\beta_{n+2} + \dots + P_{n-1}P_n\beta_M) = Y \quad \text{Equ 11}$$

où P_i est le i -ème paramètre et β_i est le coefficient de paramètre P_i . Aussi $P_i P_j$ est la multiplication des deux paramètres P_i et P_j . Le nombre de coefficients M , ce qui est plus que le nombre de paramètres n , est calculé comme suit:

$$M = \sum_{i=1}^n i \quad \text{Equ 12}$$

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

Enfin pour les échantillons S (topologies), le seuil de saturation estimée [ST] est calculé comme suit:

$$\underbrace{\begin{bmatrix} P_{11}, P_{12}, \dots, P_{1n}, P_{11}P_{12}, P_{11}P_{13}, \dots, P_{1n-1}P_{1n} \\ P_{21}, P_{22}, \dots, P_{2n}, P_{21}P_{22}, P_{21}P_{23}, \dots, P_{2n-1}P_{2n} \\ \vdots \\ P_{S1}, P_{S2}, \dots, P_{Sn}, P_{S1}P_{S2}, P_{S1}P_{S3}, \dots, P_{Sn-1}P_{Sn} \end{bmatrix}}_{[Param]_{S \times M}} \times \underbrace{\begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}}_{[\beta]_{M \times 1}} = \underbrace{\begin{bmatrix} ST_1 \\ ST_2 \\ \vdots \\ ST_S \end{bmatrix}}_{[ST]_{S \times 1}} \quad \text{Equ 13}$$

Par conséquent, le débit de seuil de saturation de l'estimation basée sur la méthode quadratique comporte quatre étapes principales:

1. Générer des échantillons (topologies):
Dans cette étape, les topologies aléatoires (f_i.topo) de Réseau sur puce 3D partiellement connectés verticalement avec le placement et l'affectation différente et aléatoire sont générées. De plus, les seuils de saturation (de f_i.st) des topologies générées sous ascenseur Première routage algorithme (chapitre quatre) [68] et ascenseur Premières routeurs (chapitre cinq) [72] sont calculées par la méthode d'analyse qui est présentée dans [71].
 2. Générer de la matrice de paramètres :
Dans cette étape , la matrice de paramètres [PARAM] est produit en analysant les topologies générées .
 3. Calculer de la matrice de coefficients (Méthode savoir la phase):
Dans cette étape, la matrice de coefficients $[\beta]$ est calculée comme Equ 14 fois sur la base de la matrice de paramètres [PARAM] et de la matrice de seuil de saturation [STr]:
- $$[\beta] = ([Param]^t \cdot [Param])^{-1} \cdot [Param]^t \cdot [ST_r] \quad \text{Equ 14}$$
4. Calculer la matrice des seuils de saturation estimés (phase de la méthode d'essai):
Dans cette étape, le seuil de saturation estimée est calculé comme suit en utilisant Equ 13 et Equ 14 :

$$[Param]_{S \times M} \times [\beta]_{M \times 1} = [ST_e]_{S \times 1} \quad \text{Equ 15}$$

Les paramètres de la matrice de paramètres [Param]

Sur la base de la méthodologie décrite et les quatre paramètres basés sur la région, nous définissons les six principaux paramètres architecturaux, comme suit, qui sont utilisés dans la matrice de paramètres [PARAM]:

- Réseau hop-count moyenne:

Le nombre de sauts moyen d'une topologie est différent pour différents modèles de trafic et des algorithmes de routage. Par conséquent, pour un modèle de trafic et le routage algorithme spécifique, ce paramètre montre l'effet de la topologie et de routage algorithme sur le temps d'attente moyen et la consommation de puissance d'un NoC. Comme, dans ce chapitre, le modèle de trafic est uniforme, le moyen nombre de sauts de réseau est la moyenne de la Manhattan houblon entre tous les nœuds de la topologie sous Elevator First algorithme de routage. Ce paramètre est à la fois sous l'influence de placement des ascenseurs et des affectations de nœuds 2D.

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

- L'écart-type de la Région-Diplôme:

Nous avons montré précédemment que si la région, Degrés de ascenseurs sont réglés sur les valeurs qui sont proches de la moyenne de la région de degré ($\frac{N}{E}$), les topologies en vertu de trafic uniforme ont un seuil de saturation élevée. Par conséquent, nous considérons l'écart-type de la Région-Degré dans le réseau d'observer si la Région-Degré de régions sont proches de la valeur moyenne ou pas et aussi son effet sur le seuil de saturation. Ce paramètre est seulement sous l'influence des affectations de nœuds 2D.

- L'écart-type des régions Hop-Count

Cela dépend de deux paramètres de placement et d'affectation. Si nous supposons que la Région-Degré idéal pour toutes les régions dans une couche est la moyenne de la région degré (RD_avg), puis la région Hop-Count idéal d'une telle région est calculé comme Equ 16:

$$RHC_{idl} = \frac{-2([h]^3 + 3[h]^2 + (\frac{N}{E} - 3)[h] + (\frac{N}{E} - 1))}{3(\frac{N}{E} - 1)} \quad \text{Equ 16}$$

où h est la nombre de sauts entre chaque noeud et son ascenseur en ascenseur de la région et calculée comme Equ 17:

$$h = \frac{-1 + \sqrt{1 + 2(\frac{N}{E} - 1)}}{2} \quad \text{Equ 17}$$

Annexe A détaille comment ces équations sont obtenues.

L'écart-type des distances moyennes Ascenseurs: Nous considérons ce paramètre pour observer la distribution des ascenseurs dans chaque couche. Par conséquent, il s'agit d'un paramètre de positionnement. Les distances moyennes Ascenseurs dans toutes les couches sont calculées Equ.18:

$$ED_{avg} = \frac{\sum_{k=1}^Z \sum_{i=1}^{R_k} \sum_{j=1}^{R_k} (|x_{elvkj} - x_{elvkj}| + |y_{elvkj} - y_{elvkj}|)}{R_t \times Z} \quad \text{Equ 18}$$

où R_t est le nombre de régions dans chacune des couches, et Z est le nombre de couches. L'écart type de la moyenne Ascenseurs distances montre comment les placements des ascenseurs dans les différentes couches sont semblables les uns aux autres.

- L'écart-type de la charge Région Degré:

Région Degré est un paramètre important qui peut montrer à peu près la charge de la Région. Si la Région-Degré d'une région est un grand nombre, cela signifie que la région de l'ascenseur doivent servir un plus grand nombre de paquets. D'autre part, les charges des éléments de la région sont très importantes. Si la Région-Degré d'une région est un grand nombre, mais les membres ne pas injecter beaucoup de paquets sur le réseau, la grande valeur de la région-Degré ne doit pas être considéré. Afin de répondre à cette question, nous définissons écart-type de charge-Région-Degré. Dans la formule de ce paramètre, la Région-de charge de chaque région est multiplié par la différence de la Région-Degré de la région et

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

la moyenne de la région, Degrés, qui agit comme un coefficient de fournir une vue plus réaliste des Régions `les performances de la topologie.

- L'écart-type de Total Région Degré (TRD_std):

Comme décrit précédemment, dans le Réseau sur puce 3D partiellement connectés verticalement les couches mises charges supplémentaires sur l'autre. Basé sur la fonctionnalité de routage ascenseur abord ces charges diminuent lorsque les ascenseurs sont placés de façon similaire dans toutes les couches, car les paquets n'ont pas besoin de houblon supplémentaires à partir du nœud récepteur à leurs Ascenseurs dans les couches intermédiaires à transmettre à leurs couches de destination. Nous définissons ainsi TRD_std, un paramètre qui est influencé à la fois par le placement et l'affectation des ascenseurs.

Résultats expérimentaux de méthode d'estimation basée sur quadratique

Dans cette section, nous présentons les résultats expérimentaux de la méthode d'estimation quadratique base. D'un point de vue pratique, pour déterminer la β_i en Equ 13, les topologies générées aléatoirement et leurs seuils de saturation sont analysées pour produire un fichier d'entrée pour Matlab, qui est utilisé pour résoudre l'équation. Tout d'abord, nous utilisons les 18500 échantillons de topologie pour rendre la matrice de paramètres [Param] et leur seuil de saturation sous motif de trafic uniforme pour faire la matrice de seuil de saturation [ST] pour les différentes valeurs d'ascenseurs dans la taille du réseau est égale à $(6 \times 6 \times 5)$, $(5 \times 5 \times 5)$, $(4 \times 4 \times 5)$, et $(3 \times 3 \times 5)$. En fait, le $(\log_{10} 18500)$ est égal à 4.26 qui est epsilon. Comme mentionné précédemment, le nombre de nœuds et des ascenseurs sont similaires dans toutes les couches et le nombre de couches Z est de 5 en permanence. Nous considérons Z constant, car il n'affecte ni placement, ni affectation.

Comme décrit précédemment, la méthode d'estimation quadratique à base comporte deux phases principales: a) apprendre la méthode de calcul de la matrice de coefficient $[\beta]$ et b) tester la méthode de calcul de l'estimée Y, qui est la matrice de point de saturation d'environ [STe]. Par conséquent, les échantillons doivent être divisés en deux ensembles, l'ensemble d'apprentissage et l'un des essais. Fondamentalement, plusieurs échantillons dans l'apprentissage de jeu font le $[\beta]$ matrice plus fiable et réduire l'erreur moyenne. Cependant, il est important de sélectionner intelligemment les échantillons. Pour cette raison, nous examinons les échantillons avec deux méthodes: 1) Sélection de l'échantillon aléatoire basé, sélection de l'échantillon sur la base priorisé, 2) Sélection de l'échantillon aléatoire basé sur 16000 échantillons sont choisis au hasard 18500 ceux d'apprendre la méthode. Dans la sélection de l'échantillon à base d'ordre de priorité, les échantillons sont choisis en fonction du pourcentage de présence de leur seuil de saturation. Nous trions d'abord les échantillons en fonction de leurs seuils de saturation et de calculer le pourcentage de leurs présences. Ensuite, nous choisissons les échantillons d'apprentissage basées sur la proportion de leur seuil de saturation:

$$\text{Proportion} = 1 - \frac{\text{STQ}}{\text{MS}} \quad \text{Equ 19}$$

CHAPITRE SIX: EXPLORATION DE L'ARCHITECTURE DE RESEAU SUR PUCE 3D PARTIELLEMENT CONNECTES VERTICALEMENT

où la STQ est la quantité de seuil de saturation de l'ensemble principal, et MS est le nombre total d'échantillons dans l'ensemble principal. Cela signifie que lorsque la quantité d'un seuil de saturation est grande la probabilité d'être choisi est faible par rapport aux autres. Ceci termine dans une sélection d'échantillons lisse parmi tous les seuils de saturation.

Nous avons mesuré les erreurs moyennes et écart type des erreurs moyennes de la taille du réseau est égale à $(6 \times 6 \times 5)$, $(5 \times 5 \times 5)$, $(4 \times 4 \times 5)$, et $(3 \times 3 \times 5)$ sont présentées. Dans tout le réseau tailles les erreurs moyennes sont améliorées (toujours environ 10%) par la réduction de nombre de Ascenseurs (quand 75% des nœuds sont Elevators). De plus, les erreurs moyennes de la méthode de sélection de l'échantillon à base de priorisé sont d'environ 5% supérieure à la moyenne des erreurs de la méthode de sélection de l'échantillon à base aléatoire.

Lorsque le nombre d'ascenseurs (E) est proche du nombre de nœuds (N), les régions sont petites. Les régions ont au plus un nœud attribué 2D à leurs ascenseurs. La plupart de la région degrés sont '1' qui signifie la région contient juste un ascenseur et pas de nœuds en 2D. Ainsi Région nombre de sauts est égale à zéro. Par conséquent, les quatre paramètres 's de la région définies ne sont pas très différents dans différentes topologies. D'autre part, lorsque le nombre d'ascenseurs est réduit, le nombre de nœuds 2D est augmenté. Cela conduit à des régions avec des nœuds plus 2D affectés à Ascenseurs. Par conséquent, les quatre paramètres de Région décrits deviennent plus gros et plus différents l'un de topologie à l'autre. L'utilisation de ce fait non seulement les caractéristiques et le comportement du réseau apparaissent clairement, mais aussi la variété des paramètres d'entrée aide de la méthode quadratique pour estimer le seuil de saturation basée sur des données plus distribués. C'est pourquoi nous croyons que l'erreur moyenne de seuil de saturation est réduite dans les deux méthodes de sélection échantillon aléatoire et hiérarchisés. Cependant, dans la méthode de sélection de priorité des échantillons, les échantillons sont sélectionnés sur la base du pourcentage de présence des seuils de saturation possibles. Par conséquent, la matrice (β) est réalisée sur la base du seuil de saturation, qui est plus probables que d'autres. C'est la raison pour laquelle l'erreur moyenne priorité de la méthode de sélection de l'échantillon est d'environ 5 % de moins que l'aléatoire.

Cependant, l'écart-type des erreurs n'est pas négligeable (environ 0.07) pour les méthodes de sélection de l'échantillon à la fois aléatoires et à base de priorité. Bien que cette méthode puisse estimer grossièrement la moyenne de seuil de saturation pour chaque (N,E) ensemble, le fait décrit montre que cette méthode doit être améliorée. Cela peut être fait par deux actions principales: 1) trouver un ensemble de paramètres qui sont plus corrélés les uns avec les autres et 2) définissent des paramètres plus architecturales qui sont plus appropriés et peuvent présenter une meilleure vue de la performance de Réseau sur puce 3D partiellement connectés verticalement. Telles sont les principales questions ouvertes et la portée d'un travail futur.

CHAPITRE SEPT:

CONCLUSION ET PERSPECTIVES

Pour surmonter le rendement et les questions de flexibilité dans les architectures 3D NoC introduites dans le chapitre deux (Définition du problème), nous avons présenté le concept de Réseau sur puce 3D partiellement connectés verticalement. Cette architecture est basée sur un système sur puce NoC empilées, dans lequel chaque couche empilée n'est pas entièrement connecté à chaque couche adjacente. Par conséquent, dans une telle architecture certains des routeurs n'ont pas UP et / ou DOWN ports verticaux. Nous avons également discuté des défis de Réseau sur puce 3D partiellement connectés verticalement qui sont 1) le routage et 2) la dégradation des performances. En outre, nous explicité que les performances en NoC est influencée à la fois par la microarchitecture (routeur) et architecturaux (topologie) des paramètres et des éléments. Par conséquent, nous avons posé trois questions principales que nous avons essayé de répondre à cette thèse:

Première: Routage dans Réseau sur puce 3D partiellement connectés verticalement

Question:

Si un nœud (routeur) ne possède pas de port UP et doit envoyer un paquet à une couche supérieure, quel chemin doit être sélectionné? En outre, si un algorithme de routage est choisi afin de déterminer un chemin de routage dans le cas, est-ce une impasse sans?

Réponse:

Nous avons répondu à cette question dans le chapitre quatre de l'explication et de l'évaluation d'un algorithme de routage sans verrouillage appelé ascenseur abord. Quand un

routeur n'a pas port UP et doit envoyer un paquet à couches supérieures, Elevator First routage transmet le paquet à la couche supérieure immédiate en utilisant un nœud intermédiaire appelé ascenseur doté d'un port UP. Un en-tête temporaire contenant l'adresse de l'ascenseur est ajouté au paquet, puis elle est transmise à l'ascenseur par X-Première routage non blocage. Pour faire face à la question de l'impasse dans Elevator First algorithme de routage, le flux de paquets va couches supérieures est complètement et physiquement séparé du flux de paquets va vers les couches inférieures. Par conséquent, deux réseaux ont été définis appelé Z+ (contient les paquets allant vers les couches supérieures) et Z- (contient les paquets allant vers les couches inférieures). Ceci assure que le cycle ne se produit jamais.

Un modèle SystemC cycle-précis de l'Elevator First routeur a été mis au point pour étudier la faisabilité de l'Elevator First algorithme de routage. Nous avons analysé les résultats de la latence moyenne d'un (5×5×5) réseau par rapport au débit d'injection de flit (charge offerte de noyaux, comme la fraction de la charge maximale possible, c'est à dire une flit par cycle) dans différentes situations pour l'utilisation de respectivement la répartition du trafic uniforme (dans lequel chaque noyau envoie des paquets de manière uniforme à tous les autres noyaux avec une même probabilité et un même taux) et un localisé (quand destinations les plus proches d'être atteint ont une probabilité plus élevée). Enfin, nous avons observé que par la réduction du nombre de nœuds dans Réseau sur puce 3D partiellement connectés verticalement, le seuil de saturation du réseau de la latence moyenne augmente. Ce qui est plus grave dans la répartition du trafic uniforme que dans un localisée. Toutefois, lorsque 10% des liaisons verticales sont éliminés les performances du réseau sous circulation régionales est approximativement la même que celle d'un réseau 3D-Mesh entièrement connecté en utilisant l'habituel Z-First algorithme de calcul d'itinéraire. En outre, Ascenseur abord routage a été comparé avec Z-First algorithme de calcul d'itinéraire dans un NoC 3D entièrement connectés avec la même taille de la mémoire tampon. Comme on s'y attendait, l'ascenseur Première routage a mieux seuil de saturation, en raison de la séparation de HAUT et BAS trafics via Z+ et Z- réseaux respectivement.

Deuxièmement: Dégradation des performances (routeur Microarchitecture):

Question:

Comme routeur microarchitecture joue un rôle clé dans la latence du réseau, ce qui est un compromis entre les avantages de l'architecture du routeur et ses coûts? Comment un routeur dans Réseau sur puce 3D partiellement connectés verticalement peut être conçu pour être aussi simple que possible, d'ajouter le moins de charge, et de manipuler l'Elevator First mécanisme de routage?

Réponse:

Nous avons abordé ces questions dans le chapitre cinq en proposant un routeur micro-architecture basée sur Elevator First algorithme de routage. En outre, nous avons développé routeur 3D Elevator First en VHDL et synthétisée avec Synopsis utilisant la technologie CMOS 65 nm de STMicroelectronics (GP65LVT). Nous avons synthétisé le routeur avec la contrainte 0ns pour atteindre la fréquence de fonctionnement maximale. Dans le chapitre cinq

, nous avons montré que la zone de gestion d'Elevator First routeur , par rapport à un routeur 7 ports équivalent avec la même taille de mémoire tampon, n'est que de 8% . Cette surcharge est due principalement à l'Elevator First FSM dans tous les ports d'entrée, un module supplémentaire de routage dans tous les ports d'entrée 2D, un module supplémentaire d'arbitrage dans tous les ports de sortie 2D, et les multiplexeurs qui séparent le Z+ et Z-réseaux.

Troisièmement: la dégradation de la performance (System-topologie):

Question:

Le mélange d'un certain nombre et le placement des Ascenseurs (liens verticaux) et leur affectation aux nœuds 2D crée une topologie en Réseau sur puce 3D partiellement connectés verticalement. Comme par réduction du nombre de Ascenseurs, seuil de saturation de latence moyenne augmente, comment le concepteur placer correctement et affecter les ascenseurs pour prévenir ou dégradation des performances au moins de la limite? En outre, il ya des paramètres tout de topologie liée ce modèle avec précision le comportement de Réseau sur puce 3D partiellement connectés verticalement? Si il ya, comment peuvent-ils montrent les changements dans la performance et pourquoi? Néanmoins, le grand nombre de possibilités de topologie est un problème grave qui peut facilement menace l'évaluation de l'exploration architecturale Réseau sur puce 3D partiellement connectés verticalement. Par conséquent, est-il une méthode simple et pratique qui peut estimer rapidement le seuil de saturation de la latence moyenne comme un indicateur de la performance?

Réponse:

Nous avons abordé ces questions au chapitre six. Tout d'abord, basée sur le concept et la structure de Réseau sur puce 3D partiellement connectés verticalement, nous avons défini certains paramètres architecturaux tels que le nombre de nœuds d'un ascenseur doit servir et les houblonnières comptes moyens entre l'ascenseur et les nœuds affectés. Ensuite, nous avons présenté une méthode de placement sur la base de modèles prédéfinis pour placer des tendances similaires sur une couche. Dans la méthode "à base de placement de motif", les ascenseurs sont placés sur chaque couche de manière uniforme. A titre d'exemple, nous avons montré un modèle avec du houblon minimum entre chaque ascenseur et ses nœuds affectés pour (5×5×5) réseau.

Les expériences des topologies de placement basée sur des modèles ont montré que lorsque la région, Degrés de toutes les régions sont à peu près similaire dans le réseau, le réseau a une meilleure performance par rapport au réseau de la région de degrés de différentes Régions. Par conséquent, nous avons utilisé la moyenne région degré ($\frac{N}{E}$) de présenter une méthode pour attribuer les nœuds des Ascenseurs égale et uniforme appelé "Affectation ascenseur uniforme". Nous avons évalué cette méthode en deux stratégies de placement différentes basées sur 1) "de placement hop-count" qui garantit minimum nombre de sauts entre les nœuds et leurs ascenseurs et 2) "coin ascenseur et le placement de bord" qui tente de placer les ascenseurs sur les coins et bords et fournir des réseaux Z+ et Z- non

chevauchement. Les résultats montrent que, lorsque E est supérieur à la moyenne région de degré $(\frac{N}{E})$, la deuxième stratégie de placement est meilleure que la première. Alors que lorsque E est inférieure à la moyenne région de degré $(\frac{N}{E})$, la première stratégie de placement est meilleur que le second. Par conséquent, nous pouvons conclure que non seulement tous les paramètres définis sont utiles, mais aussi concevoir un algorithme qui peut placer et affecter les ascenseurs sur la base des conclusions de prendre automatiquement que maille taille et le nombre des ascenseurs comme intrants seraient très pratique. En outre, une comparaison plus fiable, il est préférable de comparer les résultats de l'algorithme avec des échantillons générés de façon plus aléatoire. C'est l'un des futurs travaux de cette thèse.

Enfin, nous avons utilisé l'écart type des paramètres définis pour proposer une méthode quadratique base pour estimer le seuil de saturation de Réseau sur puce 3D partiellement connectés verticalement. En analysant les résultats expérimentaux, on constate que l'erreur moyenne de la méthode est d'environ 5%. Cependant, l'écart-type des erreurs ne sont pas négligeables (environ 0.07). Par conséquent, bien que cette méthode peut estimer grossièrement la moyenne de seuil de saturation pour chaque (N,E) ensemble, le fait décrit montre que cette méthode doit être améliorée. Cela peut être fait par deux actions principales: 1) trouver un ensemble de paramètres qui sont plus corrélés les uns avec les autres et 2) définissent des paramètres plus architecturales qui sont plus appropriés et peuvent présenter une meilleure vue de la performance de Réseau sur puce 3D partiellement connectés verticalement. Ce sont d'autres travaux futurs de cette thèse.

CHAPTER ONE: INTRODUCTION

Network-on-Chip (NoC) has been introduced initially by Guerrier et al. [14] to cope with the scalability issue in Shared-Medium interconnections within System-on-Chip (SoC) in which the number of Intellectual Property (IP) blocks was rising drastically. Some of the advantages of NoC are as follows:

- Scalability: Due to the existence of NoC routers, large number of IPs can communicate via them without any congestion and with more effective power consumption. In addition, aggregated bandwidth links scale with the network size.
- Flexibility: The design of NoC routers can be used for any network size and re-instantiated for many technologies. Furthermore, the IP blocks can be chosen among various applications (heterogeneous NoC based SoC) or similar ones (homogeneous NoC based SoC) and work with different or similar clock frequencies [13-17]. Moreover, the network can be packet-switch or circuit-switch. Of course, the former method ends in more simultaneous connection and less power consumption and, the latter method brings more resources for latency-sensitive applications.
- Transparency: The atomic and independent layers in design flow of Network on Chip lead to shorter time-to-market since adding new services to one layer may only need to modify the functionality at one layer, reusing the functions provided at all the other layers [38].
- Modularity: The routing decisions of the packets through the interconnection network whether can be made by the NoC routers in a distributed way or by a central decision making unit in the network. In the former, not only many paths can be utilized between source and destination nodes concurrently, but also all the links are not busy when they are not needed. The former ends in robust and fault tolerant design and the latter leads to less power consumption.

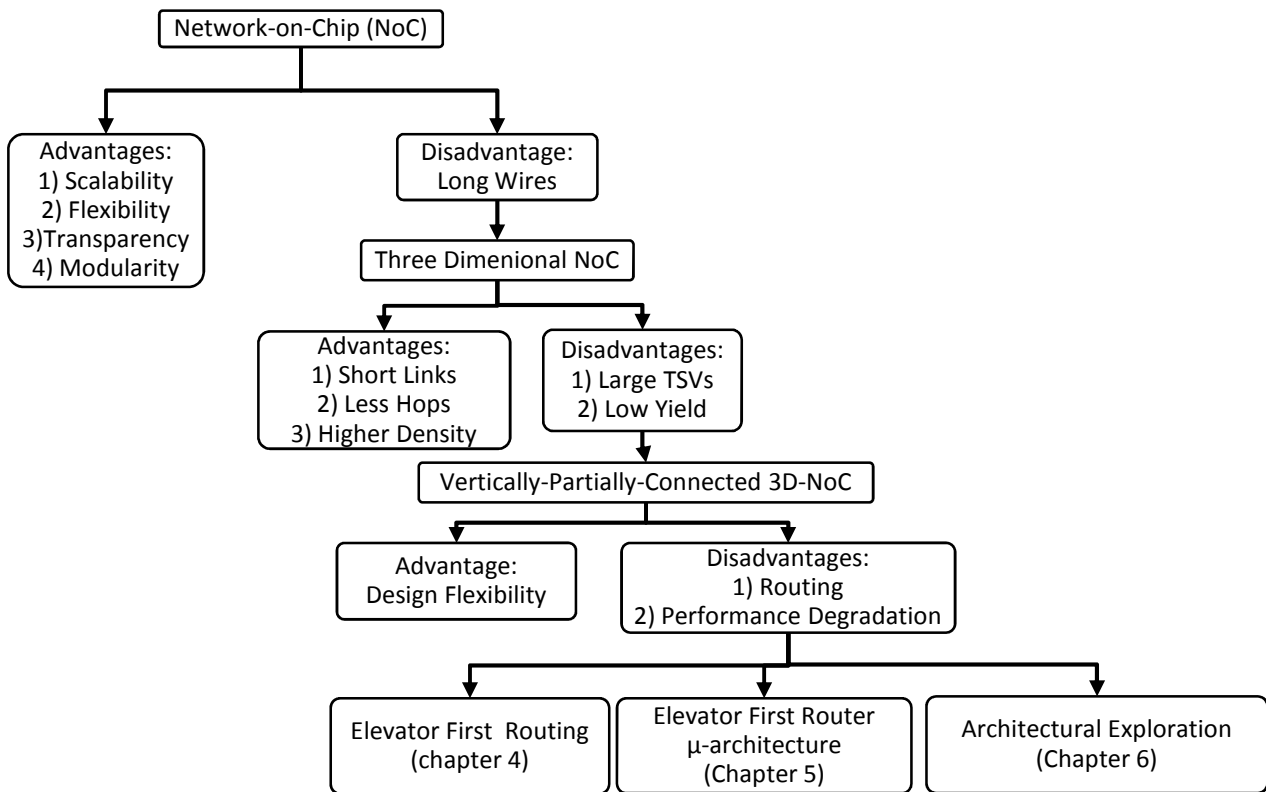


Figure 1.1: The thesis roadmap

On the other hand, Network-on-chip suffers from the delay and high power consumption of long wires and the growth of hop-count when the number of IP cores grows [44]. Utilization of the third dimension (3D technology) can lead to a significant reduction in power and average hop-count in Networks-on-Chip (NoC) [39]. The combination of 3D technology and NoC approach ends in denser integration of different on-chip layers with different technologies such as on-chip memory SoCs [42]. In addition, TSV technology, as the most promising technology in 3D integration, offers short and fast vertical links inter stacked layers [28]. Nonetheless, TSVs are huge [35] and their manufacturing process is still immature, which reduces the yield [45] of 3D NoC based SoC.

Therefore, Vertically-Partially-Connected 3D-NoC has been introduced to target both three dimensional (3D) technology and high yield. Moreover, Vertically-Partially-Connected 3D-NoC is flexible, due to the fact that the number, placement, and assignment of the vertical links in each layer can be decided based upon the requirements of the design. However, there are challenges to present a feasible and high-performance Vertically-Partially-Connected 3D-NoC due to the removed vertical links between the layers. An analysis of the pros and cons of removing some vertical links is presented Figure 1.1.

Therefore, this thesis addresses the following challenges of Vertically-Partially-Connected 3D-NoC:

1. Routing

Routing is the major problem of the Vertically-Partially-Connected 3D-NoC. Since some vertical links are removed, some of the routers do not have up or/and down ports. Therefore, there should be a path to send a packet to upper or lower layer which obviously has to be determined by a routing algorithm. The suggested paths should not cause deadlock through the network. Consequently, a deadlock- and livelock-free routing algorithm is proposed and evaluated for Vertically-Partially-Connected 3D-NoC. This problem has been addressed in [68] and in *Chapter Four* of this thesis.

2. Performance degradation

The NoC performance is affected by both 1) micro-architecture of routers and 2) architecture of interconnection.

a. Performance improvement at micro-architecture level

The router micro-architecture has a significant effect on the performance of NoC, as it is a part of packet transportation delay. Therefore, the simplicity and efficiency of the design of NoC router micro architecture are critical issues, especially in Vertically-Partially-Connected 3D-NoC which has already suffered from high average latency due to some lacking vertical links. Therefore, not only the router has to exactly and quickly transfer the packets based on the routing algorithm, but it should also consume a reasonable amount of area and power. This problem has been addressed in [72] and in *Chapter Five* of this thesis.

b. Performance improvement at architecture level

The number and placement of vertical links have a key role in the performance of the Vertically-Partially-Connected 3D-NoC, since they affect the average hop-count and link and buffer utilization in the network. Furthermore, the assignment of the vertical links to the routers which do not have up or/and down port(s) is an important issue which influences the performance of the 3D routers greatly. Therefore, the architectural exploration of Vertically-Partially-Connected 3D-NoC is both important and non-trivial. Defining, studying, and evaluating the parameters which affect the behavior of the network can be helpful to place and assign the vertical links in the layers. Finally, proposing a method which quickly estimates the saturation threshold of the network's average latency based on a few parameters is interesting and practical. This problem has been addressed in *Chapter Six* of this thesis.

CHAPTER TWO: PROBLEM DEFINITION

Nanoscale technology shrinking results in the integration of billions of transistors. Unfortunately, if the dimensions scale well other device characteristics do not i.e timing and power. Thus sequential computation centric (see Figure 2.1.a) architecture are reaching the limits the scalability is reaching thorough parallelism, thanks to the communication centric architecture (see Figure 2.1.b). several architecture templates have been define among which some have small number of complicated super-scalar microprocessors (CMP3) and other a large number of small and simple processing nodes (MPSoC4). Both share the same high performance interconnection concern even though may have different constraints (see Figure 2.2).

1. System-on-Chip (SoC)

System-on-Chip is a solution not only to avoid power hungry and slow off-chip wires between the computational cores, but also to provide more efficient parallelism among them.

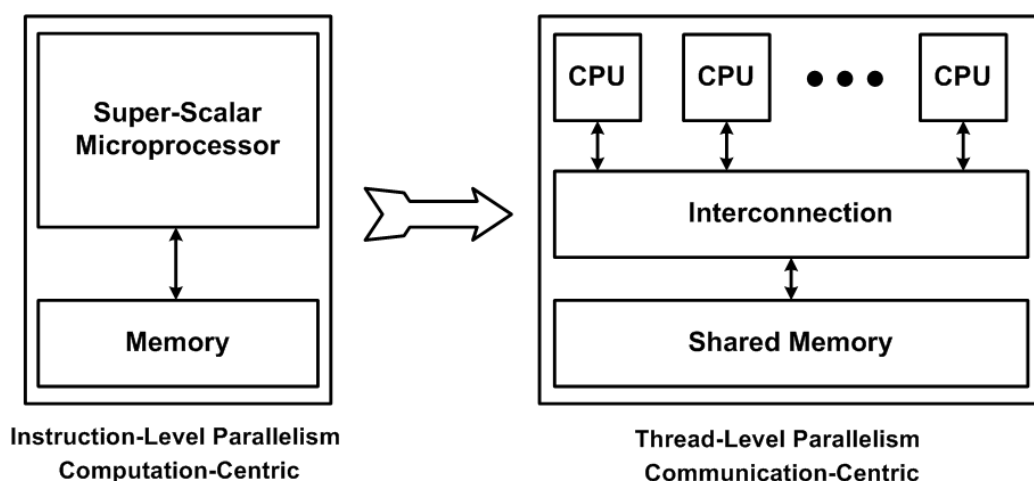


Figure 2.1: Computation-centric vs. communication-centric approach

³Chip Multi-Processors

⁴Multi-Processor System-on-Chip

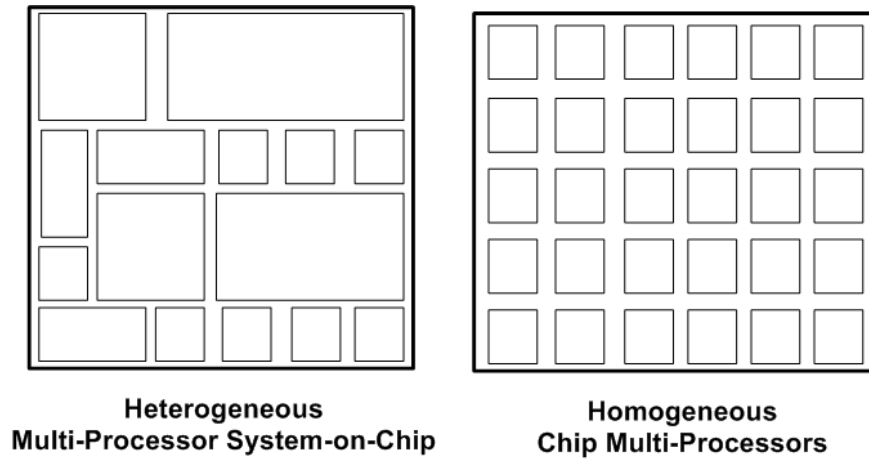


Figure 2.2: Heterogeneous and homogeneous Multi-Processor systems

As the number of cores in SoC grows, the importance of interconnection functionality increases. Picking out a scalable high-bandwidth, low-latency, and power-efficient interconnect to link the cores to each other can dramatically improve the performance of the System-on-Chip.

1.1. Shared Medium and Crossbar Interconnection

Crossbar and Shared-Medium interconnections were the primary interconnections of SoCs. Although Crossbar networks are very fast, they use a large number of links. Therefore, they are practical for high performance and small scale multiprocessing systems [3]. On the other hand, Shared-Medium networks are simple to implement, (see Figure 2.3.a). However, they suffer from contention which makes bus based interconnection impractical to use and especially for delay sensitive stream-based multimedia applications. Shared-Medium networks are not scalable (not more than 10 nodes [4][5]) which is contradictory with the philosophy of SoC (simultaneously utilizing large number of simple CPUs).

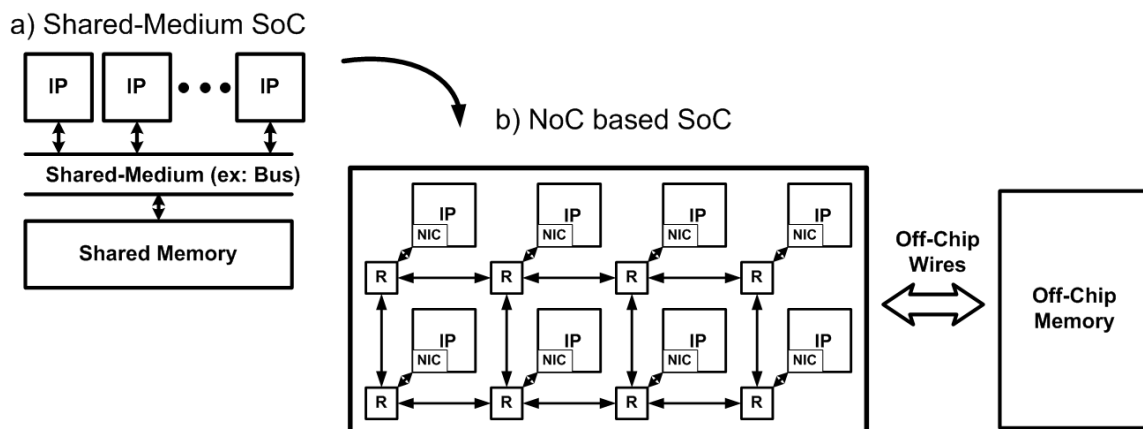


Figure 2.3: Scalable Evolution of System on Chip interconnections (from bus based SoC to NoC based SoC)

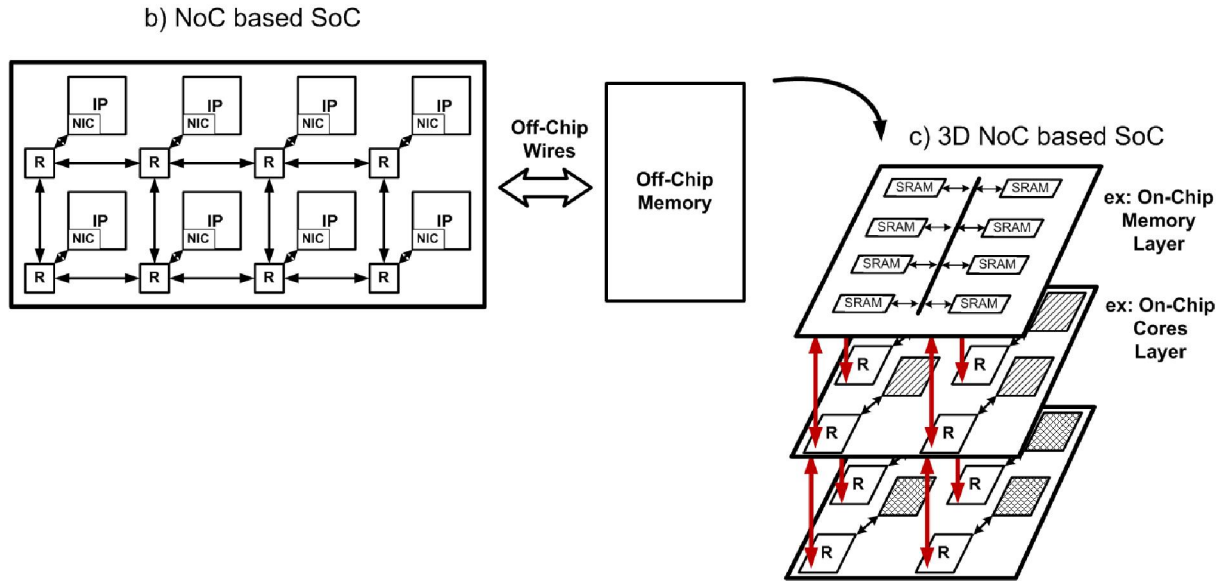


Figure 2.4: Dimensional Evolution of System on Chip interconnections (from 2D NoC based SoC to 3D NoC based SoC)

1.2. Network-on-Chip (NoC)

Network-on-Chip (NoC) is a solution for scalability and long wire problems in System-on-Chip (see Figure 2.3.b). Network-on-Chip is the emerging solution for scalable connection of IP blocks. Since its introduction in 2000 [14], many NoC architectures and micro-architectures have been proposed. These NoCs have been evaluated based on timing performance, area consumption and power dissipation along with reliability issues such as [3][18-22].

2. Three Dimensional NoC (3D NoC)

The emerging of third dimension in SoC interconnection is a result of the geometrical scaling in System-on-Chip platforms. 3D technology can sustain the growth of ICs' performance and the number of transistors (see Figure 2.4.c).

2.1. Dense Integration

The studies show that amelioration of functionality in System-on-Chip does not grow proportionally with the technology shrinking anymore [23-25]. By scaling transistor's size, leakage and parasitic increase which causes more power consumption. 3D technology provides a very dense integration and a possibility to reuse different technologies and designs. The vertical links in 3D NoC, which link different layers with billions of transistors to each other, provide a possibility to compact integration which was seriously threatened by the problems of deep sub micron technology shrinking.

2.2. Lower Average Latency and Power Consumption

In addition, as customers need to execute more applications in parallel, more on-chip computational cores are required. This enlarges the length of communication path which means more clock cycles and growth of network latency. Also, the horizontal expanded

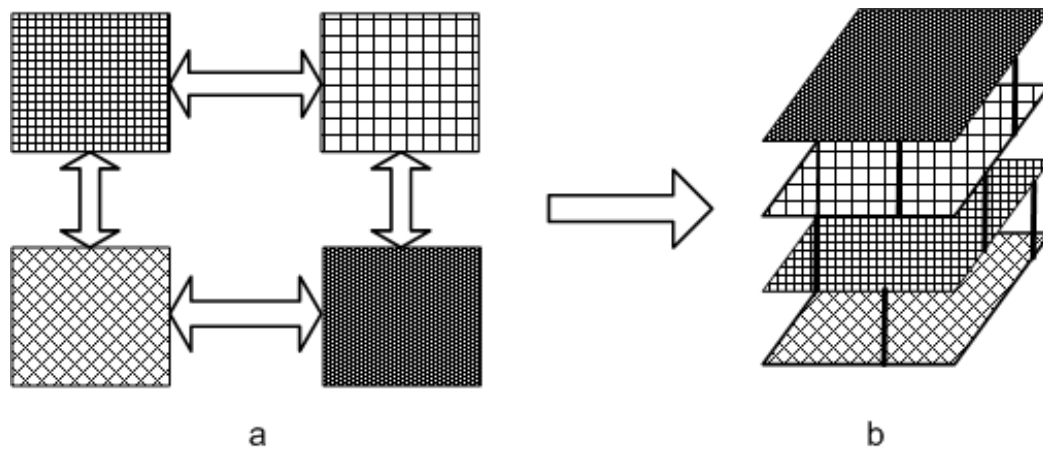


Figure 2.5: Off-chip links vs. On-chip links

interconnection significantly increases power consumptions of the communications. 3D NoC resolves these problems by expanding the on-chip network in the third dimension reducing the network's diameter, which ends in smaller hop-count. Due to the fact that, the length of on-chip inter layers vertical links is much shorter and the width of them is much larger and less power hungry than horizontal links [28]. In other words, 3D NoC limits the problem of long diameter in 2D NoC and improves performance. Some works such as [74] propose optical 3D NoC-based SoCs to cope with the problem of electrical interconnect by increase immunity to electromagnetic noise. In addition, optical 3D NoC architectures improve bandwidth, reduce latency, and decrease power consumption. Although this could be an interesting solution in the future, we shall focus on electrical interconnect in this thesis.

2.3. Flexible Integration and Feasible On-Chip Memory

Additionally, off-chip links are very slow and eat up considerable amount of power and their numbers are restricted to the chips' number of pins [24] (see Figure 2.5.a). Therefore, not only they are not cost-efficient, but also their bandwidths do not commensurately grow with the functionality of the chips. This problem shows itself severely in off-chip links between MPSoC and memory (DRAM) blocks [27]. 3D NoC based SoC creates a feasible opportunity for a flexible integration. The possibility of stacking different dies with different technologies, size, applications, and clock frequencies in 3D NoC alleviates these problems and provides a flexible integration (see Figure 2.5.b).

3. Through-Silicon-Vias (TSVs)

Through-Silicon-Vias (TSVs) is the most dense, practical and promising wafer-level technology of vertical links in 3D intergradations. This is a very useful achievement which dramatically reduces the interconnection latency and increases design flexibility. Moreover, TSVs are 50 times shorter and 10 times faster than intra layer wires [29], which all of the described features also end in less power consumption. Although, 3D is a viable integration method, there are some challenges such as TSV size and the relation of yield ratio and the number of TSVs.

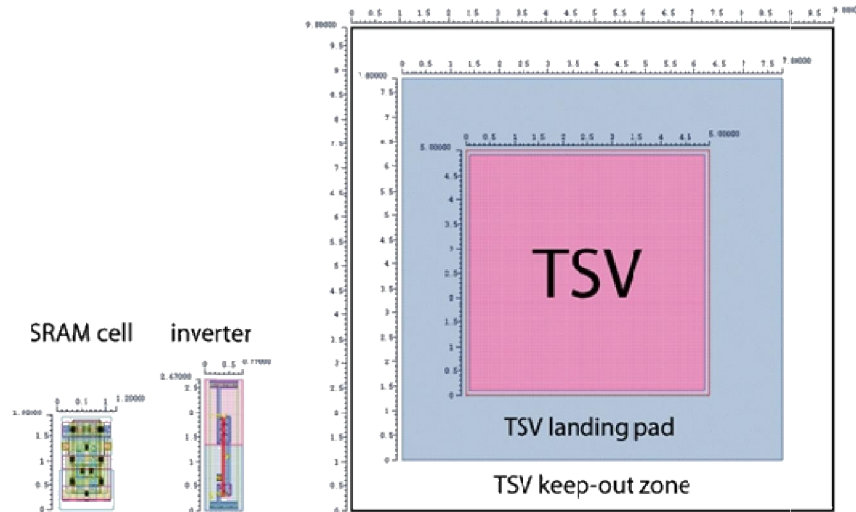


Figure 2.6: TSV with landing pad and keep-out area is several times larger than gates and memory cells [35]

3.1. TSV Area Consumption

One of the challenges of 3D integration is the surface of TSV's pad. As it is shown in [35] and [24], TSV's landing pad and keep-out area is considerably large, as can be seen in Figure 2.6. Usually some extra circuits are required for electrostatic discharge protection and signal level adaptation may also be needed. According to this, by utilizing of more TSVs, although the delay reduces significantly, the area consumption of interconnection increases dramatically. Therefore, not only less free space on each layer remains for logic and wires, but also the length of horizontal wires increases. Furthermore, the alignment precision of TSVs in different layer is very important. Although ad-hoc placed TSVs are very efficient for timing and power issues, fabrication of ad-hoc placed TSVs is complex from the technology standpoint and might not be easily manufacturable. This is the reason why albeit regular placement of TSVs is not very efficient, it is sometimes more economical and has lower time-to-market to use them compared to irregular placement. All in all, the placement of TSVs is an important issue.

3.2. Yield vs. Number Of TSVs

Another important issue in 3D integration is the number of TSVs. As can be seen in Figure 2.7, the yield ratio is decreased by the growth of the number of TSVs. Of course, the value of yield ratio reduction in various technologies is different. It is based on the sensitivity and features of the TSV manufacturing technology. However, in all the technologies which are demonstrated in Figure 2.7, yield declines (defect rises) in an exponential manner when number of TSVs is more than 1000 (IMEC) or 10000 (IBM and HRI-JP).

Likewise as estimated in ITRS 2009 [36], design process and CAD tools of three-dimensional chip stacking through the usage of high density TSVs is a key focus area to address latency and power concerns and will be improved. Nevertheless, the 3D integration can be more useful when a balance between reasonable number of huge and unreliable vertical links (TSVs) and benefits of them has been found. In other words, not only we have

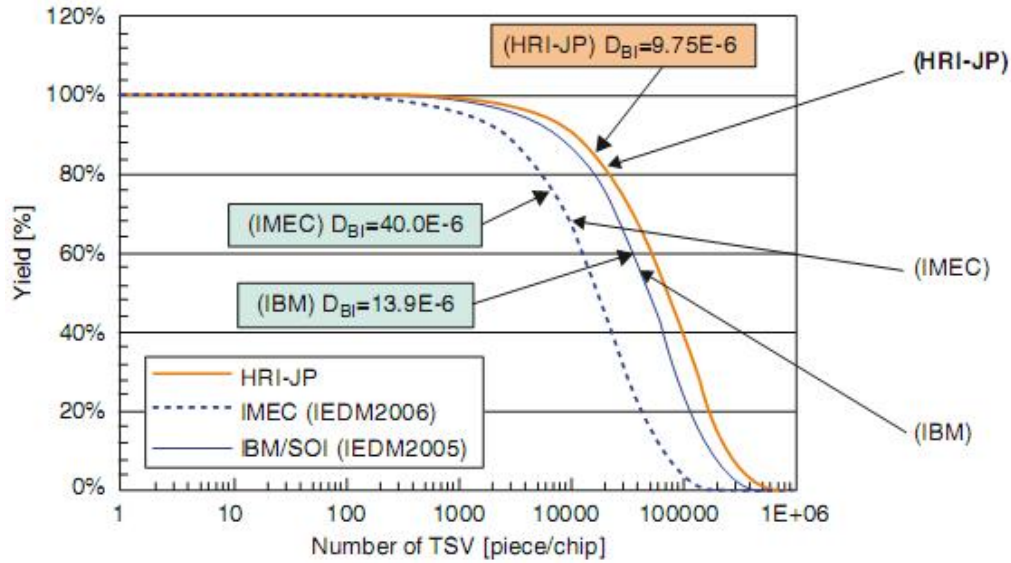


Figure 2.7: Yield vs. TSV count [45]

to take advantage of short, fast, and flexible TSVs, but also we have to have an acceptable yield and sufficient area for intra layer logic and interconnection.

4. Vertically-Partially-Connected 3D-NoC

Vertically-Partially-Connected 3D-NoC is a stacked NoC based SoC. Figure 2.8.a depicts a three layers example of Vertically-Partially-Connected 3D-NoC. As can be seen in Figure 2.8.b, in this architecture the NoC routers are divided into two categories: a) 2D Routers and b) 3D Routers. 2D Routers are the conventional 5 ports (N, W, S, E, L) routers which are not able to receive or send packets from/to lower or upper layers. While, 3D Routers have two Up and Down ports which let them receive and send packets from/to weather upper and/or lower layers, based on their type. 3D routers in Vertically-Partially-Connected 3D-NoC are categorized into three categories: a) 3D routers that have only UP port called “3D Router-UP”, b) 3D routers that have only DOWN port called “3D Router-DWN”, and c) 3D routers that have both UP and DOWN ports called “3D Router-UP/DWN”. Furthermore, data flow in vertical links in Vertically-Partially-Connected 3D-NoC can be different, as shown in Figure 2.8. As an example in Figure 2.8.b, two 3D Routers-UP/DWN in layer 2, although the type of both routers is the same, their data flows are different. Of course, this adds more flexibility to Vertically-Partially-Connected 3D-NoC and makes it more feasible and manufacturable. Therefore, we can conclude that a Vertically-Partially-Connected 3D-NoC has the following characteristics:

- Size: Number of nodes in each layer can be different.
- Number, Placement, and Assignment of TSVs: Number and placement of TSVs between two adjacent layers can be different. Assignment of the TSVs to the 2D nodes inside a given layer has no technological constraint and is therefore arbitrary.

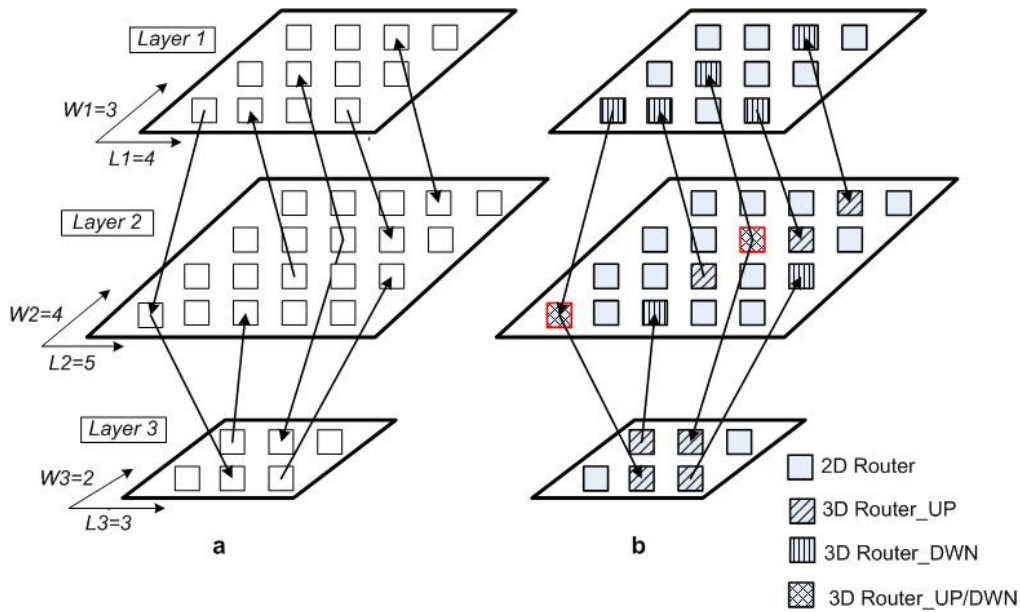


Figure 2.8: a) An example of Vertically-Partially-Connected 3D-NoC, b) NoC routers category in Vertically-Partially-Connected 3D-NoC

5. Issues in Vertically-Partially-Connected 3D-NoC

Although Vertically-Partially-Connected 3D-NoC can create a very dense, flexible, and manufacturable infrastructure for SoC, some challenges still exist due to the removed TSVs:

- Routing in Vertically-Partially-Connected 3D-NoC
- Degradation of performance due to vertical bandwidth reduction

In this thesis, we categorized the problems based on system and architectural points of view. In [2], Network-on-Chip has been divided into two main architecture and system parts. As interestingly mentioned in [2], not only these two categories cover all architectural issues of NoC, but also they complement each other:

- System-Level: This part contains Topology and Routing Mechanism and covers the infrastructure of the nodes and the connectivity protocols between them.
- Architecture-Level: This part includes Routers RTL⁵ Architectures and Network Interfaces (NI) and covers the detailed components within a network node.

As mentioned before, interconnection plays a key role in NoC based SoC. Since, “different interconnections” bring about “different packet routing algorithms” which lead to “different router micro architectures”. So, they are tightly knitted to each other. Correspondingly NoC connectivity (packet routing), performance (network latency), and cost (area and power consumption) are sensitively affected by each other. Moreover, complicated routing algorithms can minimize the hops in the paths between the nodes and avoid

⁵ Register-Transfer Level

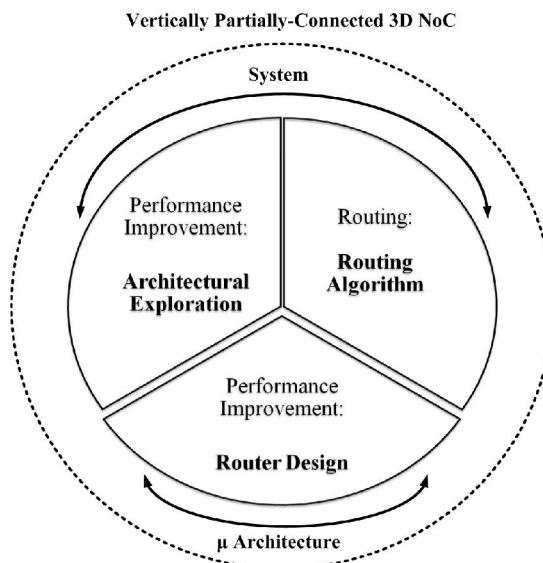


Figure 2.9: Vertically-Partially-Connected 3D-NoC challenges

contentions in the links. These complexities in routing end in more control logics in router micro architectures which finally lead to more area and power consumption. Therefore, close relationship between interconnection, routing algorithm, and router architecture requires designers to find tradeoffs in the design parameters to fit a given application/purpose.

Based on the described high-level overview of NoC, we summarize the challenges of Vertically-Partially-Connected 3D-NoC in Figure 2.9. As mentioned before, we believe that routing and performance degradation are the most challenging problems of Vertically-Partially-Connected 3D-NoC. Routing should be addressed at system level. An efficient deadlock free routing mechanism is needed to determine the paths the packets should follow, especially between the nodes which do not have any vertical links, as it is shown in Figure 2.10. Unlike routing, performance degradation should be controlled at both system (topology) and architecture (router) design levels, since the packet transport latency is the sum of both link and router latency. Therefore, performance can be kept at a desirable level with only well-distributed vertical links and low-latency 2D and 3D routers.

5.1. Problem 1: Routing in Vertically-Partially-Connected 3D-NoC

In fully-connected 3D NoC, all the routers in all the layers are connected to upper or lower layers. In other words, all routers are 3D routers. Therefore, routers can use, for example, a deadlock-free dimension-order routing algorithm like Z-First, when they need to send a packet to an upper or lower layer. As can be seen in Figure 2.10, if the node ‘S’ wants to send a packet to the node ‘D’, firstly the packet should be forwarded to upper layer via the Up port of router ‘S’ and then by deadlock free and livelock free X-First (XY) routing algorithm [37], it will be sent to router ‘D’ as its destination node.

On the contrary, in Vertically-Partially-Connected 3D-NoC, some routers (nodes) are 2D routers and they do not have any vertical links (TSVs). Figure 2.11.a presents a fully connected 3D NoC, Figure 2.11.b shows the same topology with two removed vertical links.

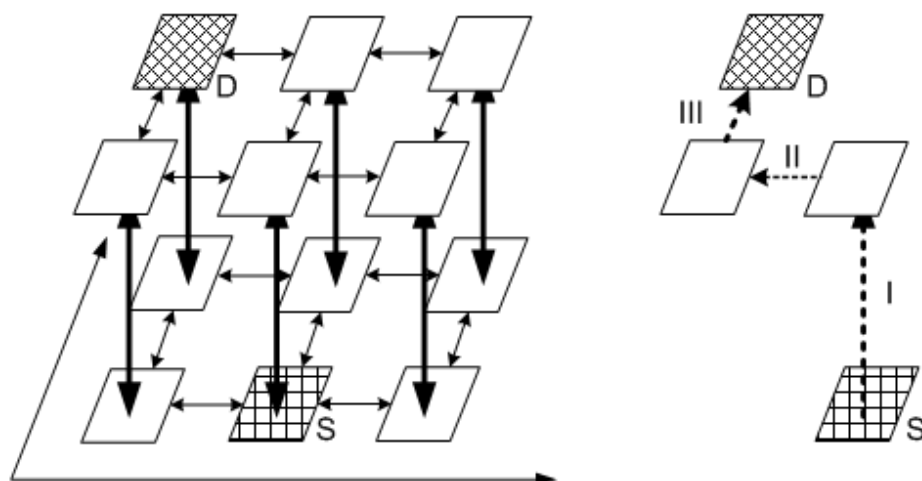


Figure 2.10: Deadlock-free Dimension order Z-First routing algorithm for Fully-Connected Mesh-based 3D-NoC

Therefore, as can be seen in Figure 2.11.c, a question can be raised. Which path the packet sent by the 2D nodes (dotted squares) should follow, when they need to communicate with another node which is located in upper or lower level? Another issue is deadlock freeness. The suggested paths should not create deadlock through the network. Therefore, a deadlock and live-lock free routing algorithm is required.

5.2. Problem 2: Performance Degradation in Vertically-Partially-Connected 3D-NoC

The second main concern in Vertically-Partially-Connected 3D-NoC is performance diminution. Lack of some TSVs in Vertically-Partially-Connected 3D-NoC means less channels or, in other words, less bandwidth. This problem becomes even worse for latency-sensitive applications. As can be seen in Figure 2.12, packet transport latency, which can be considered as a performance indicator, is the sum of router latencies (L_r), which depends on the micro-architecture (see Equ 1).

$$\text{Transport Latency} = \sum_{i=1}^n (L_{ri}) \quad \text{Equ 1}$$

where in Equ 1, n is the number of hops between the source and destination nodes, which

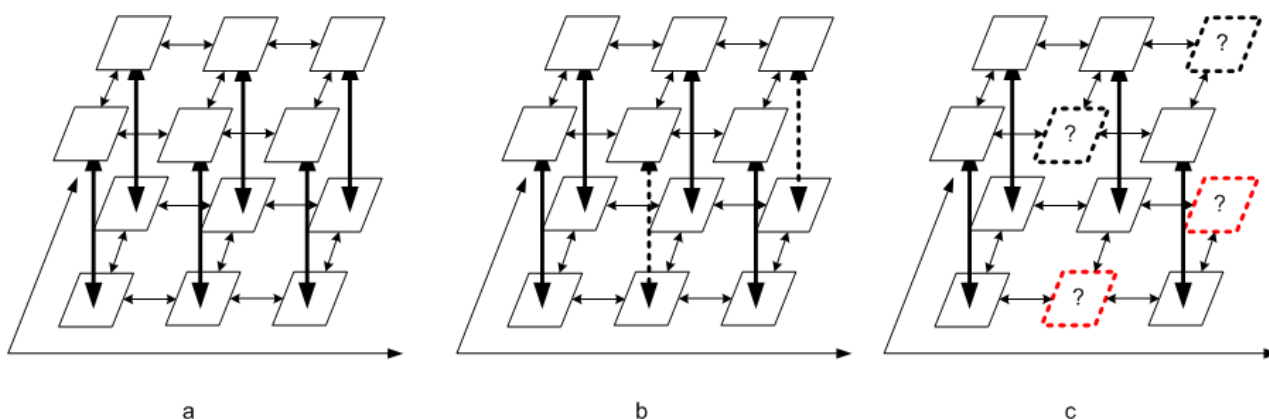


Figure 2.11: Routing problem in Vertically-Partially-Connected 3D-NoC



Figure 2.12: Transport latency

depends on the system architecture. The number and placement of vertical links significantly influence the average hop-count of topology which finally ends in significant effect on the average latency, as a performance indicator too.

Therefore from a system point of view, the “number”, “placement”, and “assignment” of vertical links affect the performance significantly; and from the micro architecture point of view the efficiency of router design influences performance. Thereupon, if these issues are well solved based on the traffic pattern and design limitations, not only the performance can be saved, but also the final architecture will be more cost-efficient due to less vertical links (TSVs) which leads to more area in intra layers and better yield.

5.2.1. Micro-Architecture Level Analysis

Besides the links and their interconnection structure, router architecture plays a key role in network latency. The complexity of router is a double-edged sword. On the one hand, complex control modules can increase reliability, avoid contention, and etc which is very valuable and sometimes necessary; and on the other hand, they can easily increase packet transport latency by adding some extra cycles for more calculations or sending and receiving some control packets. Also it may force the router to require more area and power. Therefore, finding a tradeoff between the advantages of router architecture and its costs is a complex subject which has to be taken into account. In the Vertically-Partially-Connected 3D-NoC context, not only the router design should be as simple as possible and adds extra load as less as possible, but also it should handle the routing mechanism based on the routing algorithm efficiently.

5.2.2. System Level Analysis

Fundamentally, the minimum number of vertical links (TSVs) is specified based on different factors such as acceptable yield, requirements of applications, and etc. On the other hand, in Vertically-Partially-Connected 3D-NoC, by different places of vertical links (Placement) or/and different assignments of 2D Routers (nodes) to the vertical links (Assignment) a new interconnection (Topology) is made. Therefore if we consider the topology of each layer is Mesh-based, the number of possible topologies (T) for a Vertically-Partially-Connected 3D-NoC is calculated as follows:

$$T = (P \times A)^Z \quad \text{Equ 2}$$

where P is the number of possible placements, A is the number of possible assignments, and Z is the number of layers. Clearly, if the number of placements and assignments increase, the

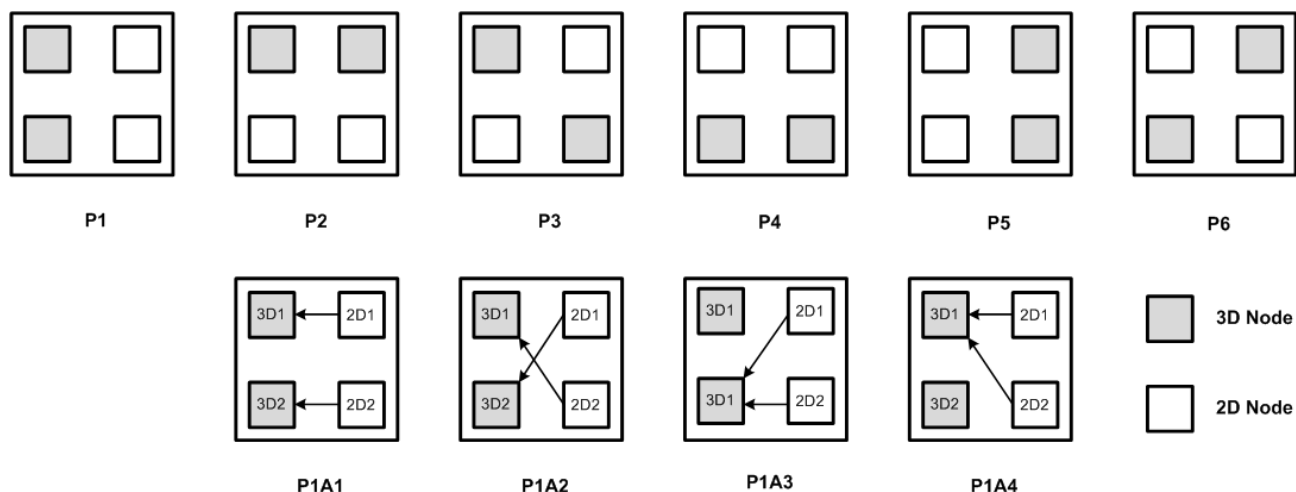


Figure 2.13: An example of possible placements and assignments for two 3D nodes and four nodes in a layer

number of possible topologies will be increased too. If we assume that the layer size, which is the number of nodes ($N = X \times Y$) in each layer, and the number of vertical links (VL) in each layer are the same in all the layers, the number of possible placements (P) and assignments (A) in each layer can be calculated as following:

$$P = \binom{N}{VL} = \frac{N!}{VL! \times (N-VL)!} \quad \text{Equ 3}$$

$$A = VL^{(N-VL)} \quad \text{Equ 4}$$

The possible placements in each layer is the combination of 3D nodes (the nodes which have vertical link) from the total number of nodes since the selection's order of 3D nodes are not important (see Equ 3). For example in the first row of the Figure 2.13, the six possible placements of two 3D nodes from four nodes in a layer are shown, as $\binom{4}{2}$ equals six. In addition, the four possible assignments of the first placement are shown in the second row of the Figure 2.13, as $(2^{(4-2)})$ equals four. As can be seen in the shown assignment possibilities, each of 2D nodes (2D1 and 2D2) can be independently assigned to two 3D nodes (3D1 and 3D2). Therefore, the total possible assignments are the multiplication of possible assignments of each 2D node.

In Figure 2.14 plots the number of possible topologies (T) for N equals to 9, 16, 25 and 36. As can be seen the number of possibilities are huge, even for ($N = 9$), which makes the analysis of topologies' behavior difficult and somehow impossible. This problem becomes worse by an increase in the size of layers (N) and reduction of number of vertical links(VL). Although by stacking in more layers average hop-count is reduced and integration becomes denser, the heat dissipation issue gets worse which lead to performance degradation and unreliability and undependability. Therefore, we believe that the size of dimension in layers will be bigger than the number of layers in 3D NoCs.

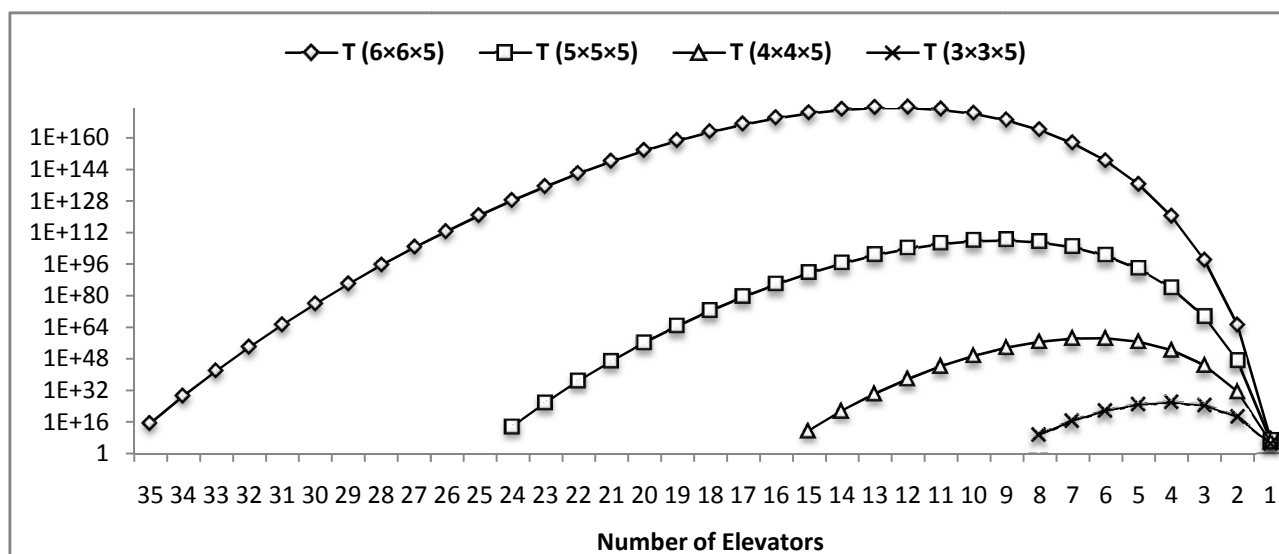


Figure 2.14: The number of possible topologies ($T = (P \times A)^Z$ for N equals to 9, 16, 25 and 36 presented in logarithm scale

Based on this analysis we can see that the architectural exploration of Vertically-Partially-Connected 3D-NoC is both critical and non-trivial. Placement and assignment of vertical links through the layers and the used routing algorithm directly affect the distribution of traffic in 3D NoC. If it is done properly, congestion is reduced, hot-spots can be avoided, and finally the saturation threshold of network will be increased. Moreover, proper assignment of vertical links to 2D nodes can end in fairly usage of vertical links and buffers in 3D routers, which means fewer blockages in the network.

Consequently, the question raised here is: how can the designer properly place and assign the vertical links to prevent or at least limit performance degradation? Furthermore, are there any parameters that by them the behavior of the Vertically-Partially-Connected 3D-NoC can be shown or measured? If there are, how can they show the changes in performance and why? Nevertheless, the huge number of topology possibilities is a serious problem which can easily threat the evaluation of architectural exploration in Vertically-Partially-Connected 3D-NoC. Consequently, proposing a simple and practical method which quickly estimates the saturation threshold of average latency as a performance indicator would be desirable.

6. Conclusion

In this chapter, the evolution and challenges of Vertically-Partially-Connected 3D-NoC have been explained. TSV technology, as the most promising technology in 3D NoC based SoC, offers short and fast vertical links. Utilization of TSVs can lead to a significant reduction in power and latency. Nonetheless, TSVs are huge and their manufacturing process is currently reduces the yield of 3D NoC based SoC. Therefore, Vertically-Partially-Connected 3D-NoC has been proposed. The Vertically-Partially-Connected architecture not only benefits from the advantages of 3D technology, but also utilizes the TSVs efficiently. This means more area for inter layer logics and interconnection and more reliable 3D ICs. Vertically-Partially-Connected 3D-NoC is very flexible, due to the fact that the number and

placement of the vertical links in each layer can be decided based on the design limitations and requirements. However, there are some challenges to present a feasible and high performance Vertically-Partially-Connected 3D-NoC due to the removed TSVs between the layers.

- Routing in Vertically-Partially-Connected 3D-NoC

This is the major problem of the Vertically-Partially-Connected 3D-NoC. Since some vertical links are removed, some of the routers do not have any up or/and down ports. Therefore, there should be a path when they want to send a packet to upper or lower layer, which obviously has to be decided by a routing algorithm. Moreover the suggested paths should not cause deadlock in the network. Consequently, a deadlock and live-lock free routing algorithm should be suggested. We will address this issue in Chapter four.

- Performance degradation in Vertically-Partially-Connected 3D-NoC

The NoC performance is affected by both system (topology and routing mechanism) and architecture (network interface and router architecture) points of view.

1. Performance Improvement at Micro-Architecture Level

The router architecture also has a significant effect on the NoC performance, as it is a part of transportation delay. Therefore, the simplicity and efficiency of the design of NoC router architecture are very important. Not only the router has to transfer the packets based on the routing algorithm, but also it should be fast and consumes reasonable amount of area and power. We will address this issue in Chapter Five.

2. Performance Improvement at System (Architectural) Level

The number and placement of vertical links have a key role in the performance of the Vertically-Partially-Connected 3D-NoC, due to the fact that they affect the average hop-count of the network. Furthermore, the assignment of vertical links to the routers which do not have any up or/and down port(s) is very important and influences the performance as well. This problem becomes worst when of the number of nodes grows in the layers, and for certain number of vertical links. Therefore the system exploration of Vertically-Partially-Connected 3D-NoC is very important and hard. Not only we should suggest a method to improve the performance by appropriate placement and assignment of vertical links, but also the method should be practical. We will address these problems in Chapter Six.

CHAPTER THREE: STATE OF THE ART

In the previous chapter, the problems this thesis addresses were explained. In this chapter the previous studies targeting each problem will be presented. The NoC-based SoC, as an inevitable approach for the future multiprocessors, is a very interesting research area for researchers. Therefore, the concepts of NoC have been explained in many studies. Duato et al. [3] and Dally et al. [37] extensively studied the elaboration and details of different interconnections but not specifically targeting VLSI integration. The concepts and principles of NoC were introduced in Guerrier et al. [14] and popularized in Benini et al. [17]. Furthermore, Nicopoulos et al. [2] evaluated the issues of NoC from system and micro architectural points of view. With the emerging of third dimension in NoC as a viable solution, Sheibanyrad et al. [24], Tan et al. [25], Vasilis et al. [39], Xie et al. [40], and Papanikolaou et al. [41] introduce the concepts, principles, and challenges of 3D NoC-based SoC. In following, the most related studies about the 3D technology and NoC integration and its advantages compared to 2D NoC will be explained:

Li et al. [42] show one of the first and comprehensive design and implementation of 3D technology and NoC integration. They consider the design of a three dimensional architecture for None-Uniform Cache Architecture which merges the advantages of NoC and third dimension to alleviate the L2 cache latencies in multi-processor systems. They also explore the challenges of managing 3D CMPs together with L2cache design space issues. Li et al. [42] select a combination of buses and NoC based interconnection to devise a communication infrastructure between CPUs and L2 caches layers. The performance of this fusion is evaluated using standard CPU benchmarks. However, this analysis only relates to homogeneous chip multiprocessors and does not consider the use of 3D network structures for application-specific SoCs.

Pavlidis et al. [43] comprehensively study about the 2/3D ICs and 2/3D NoC concepts. In addition, they show and evaluate different topologies of 3D NoCs. They also propose an analytical model to analyze the 3D NoCs. The authors consider Mesh topologies and its zero load latency modeling. In this work the processing elements (PEs) can be integrated either on a single physical plane (2-D IC) or on several physical planes (3-D IC). Finally they make the

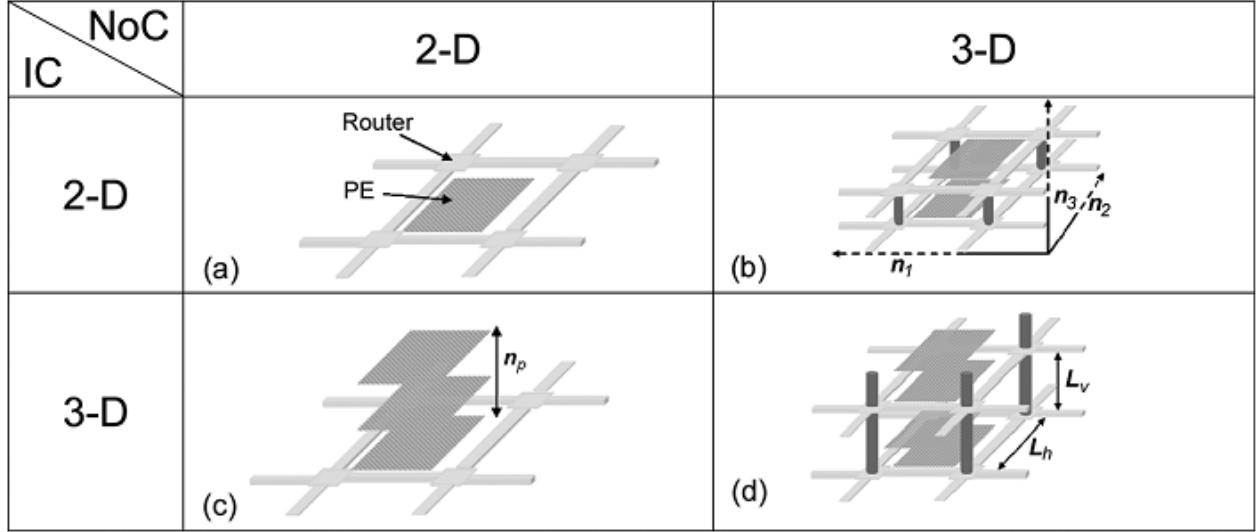


Figure 3.1: Various NoC topologies (not to scale).
(a) 2-D IC-2-D NoC. (b) 2-D IC-3-D NoC. (c) 3-D IC-2-D NoC. (d) 3-D IC-3-D NoC [43]

permutation of different types of both (2/3D IC) and (2/3D NoC) and evaluate the four types of NoCs which are shown in Figure 3.1. The authors assume that in 3D NoC, the layers are fully connected to each other. Furthermore, they compare 2D Mesh structures with their 3D counterparts by analyzing the zero load latency and power consumption of each network. Finally, they show that the performance enhancements can be achieved in NoC by utilizing the third dimension and each of the proposed 3-D topologies decreases the zero-latency of the network by reducing the hop-counts between the nodes and internal elements. They also demonstrate that although the number of routers ports goes from 5 to 7, consequently increasing the arbitration and switching delays, a short vertical bus design intra layers can compensate this.

Pande et al. [12] and Feero et al. [44] are two widespread studies respectively about 2D and 3D NoC-based SoC in terms of cost and performance indicators. Pande et al. [12] introduce an analyzing method to compare the throughput and packet latency with area and energy consumption of different 2D NoC topologies, (a) Fat-Tree, (b) CLICHÉ, (c) Torus, (d) Folded torus, (e) Octagon, (f) BFT. Feero et al. [44] extend this comparison to 3D Mesh based NoC and 3D Tree-based NoC topologies. Finally the authors show the advantages of 3D NoC topologies to 2D ones, with both having the same number of nodes. Both average hop-count reduction and performance improvement have been shown in 3D NoC topologies. Also, the authors demonstrate that with a reasonable area overhead, the 3D NoC interconnections gain a significant improvement in energy, latency, and throughput. Although Li et al. [42] show some of the advantages of 3D NoCs, the challenges of vertical bus-based links in term of the growth of number of layers have not been studied. Thanks to Feero et al. [44] which evaluate the effect of the growth of layers on network energy in different types of 3D NoC (3D non bus-based Mesh, 3D bus-based Mesh, 3D ciliated Mesh, 3D BFT, and 3D Fat-Tree), we now know that by growth of the number of layers the packet energy is reduced due to the growth of the number of vertical links. Feero et al. [44] also show that although Stacked-Mesh 3D NoC has less average hop-count than 3D Mesh, it suffers from significant

energy consumption due to the capacitive loads on the buses. Therefore, bus design can be a deterrent factor in 3D NoC design.

Finally, Seiculescu et al. [45] Shan et al. [46] Pingqiang et al. [47] study the concepts, design flow, and synthesis algorithm of application-specific 3D NoC based SoC. For example, Seiculescu et al. [45] present a synthesis tool called SunFloor3D, to synthesize application-specific 3D NoCs. The tool decides the best NoC architecture for the application, discovers paths for the data stream, assigns the network components to the 3D layers and carries out a placement of them in each layer. Several SoC benchmarks have been utilized to show a comparative study between 3D and 2D NoC designs. Finally the authors demonstrate large improvements in interconnect power consumption and delay for the 3D NoC when compared to the corresponding 2D implementation.

Until now, some of the most referred studies about the design of 3D NoC-based SoC have been described. From now on, studies about the problems that this thesis address, will be presented. Since routing and performance improvement in system and micro architectural levels in Vertically-Partially-Connected 3D-NoC are the main problems of this thesis, the following studies are categorized into three main sections:

1. Vertically-Partially-Connected 3D-NoC Routing Algorithm (Chapter Four)
2. Vertically-Partially-Connected 3D-NoC Router Micro-Architecture (Chapter Five)
3. Vertically-Partially-Connected 3D-NoC Architectural Exploration (Chapter Six)

1. 3D NoC Routing Algorithm Related Works

In Network-on-Chip, the routing module, as the main part of routers, leads the packets to their destination. This ends in a very sensitive effect on traffic distribution, and consequently on power consumption, aggregate throughput, and latency through the network. Due to the emergence of 3D NoC-based SoC, many routing algorithms have been proposed to improve the performance. Particularity in 3D NoC-based SoC, some routing algorithms

1. transmit the packets via the layers near the heat sink to alleviate the temperature [48].
2. tolerate the congestion and bus failure especially in bus based 3D NoC [49].
3. adapt themselves to the changes of topologies due to reconfigurability [53] in 3D NoC.
4. solve the disconnectivity problem of vertically irregular 3D NoC due to
 - a. Virtualization [51].
 - b. TSVs fault [50] [51] [52].
 - c. Reconfigurability [53].
 - d. Yield and TSV's pad area [54].

Matsutani et al. [48] propose a deadlock-free routing algorithm for a special type of 3D NoC topologies called Xbar-connected Network-on-Tiers (XNoTs), which consist of multiple network layers. The nodes are connected to each other horizontally and via “tier routers” and vertically fully via “pillar routers” which are cross bar switches (see Figure 3.2). The intra layer interconnection is (fully or partially and regular or irregular) determined by the

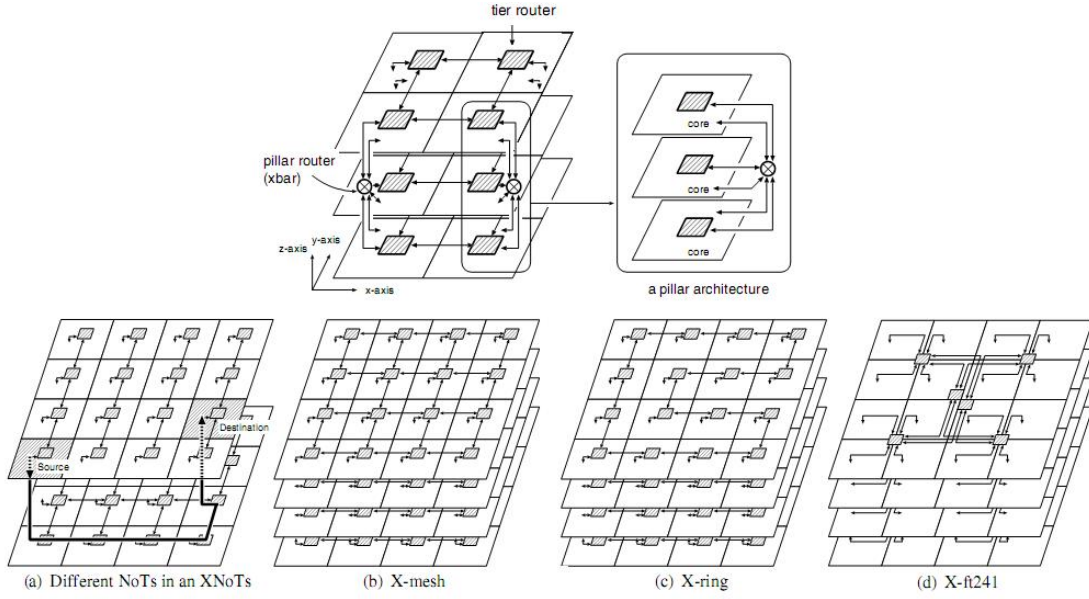


Figure 3.2: Examples of XNoTs topologies (a-d), proposed routing and a generic XNoT architecture [48]

designer, as can be seen in Figure 3.2 (a-d). Based on the intra layer interconnection, a dimension order deadlock free routing is also chosen for intra packet routing. However the bottom layer (Network-on-Tiers) has to guarantee reachability between all cores. The example in Figure 3.2.a shows that although the upper layer is not fully connected. Therefore, the packet is transmitted via the bottom layer next to the heat sink to the packet's destination (x,y) coordinates by X-First 2D routing and then forwarded to the upper layer. In order to do this their proposed routing algorithm selectively employs the paths at the bottom layer close to the heat sink of the chip, in order to mitigate the heat-dissipation problem of 3-D ICs (see Figure 3.2.a).

Their routing algorithm is deadlock free because no dependency occurs as follows. 1) No cyclic dependency is formed in each tier, because a packet must follow a 2D dimension order deadlock free routing, as long as the packet is transferred on a single tier. 2) No cyclic dependency is formed across the layers, because a packet is passed between layers only in the descending order. 3) No cyclic dependency is formed within a pillar, because a pillar router is a crossbar switch. Therefore, inter layer routing is deadlock free by prohibiting the packet transfer from a lower-numbered NoT to a higher-numbered NoT, unless the next hop is a pillar router directly connected to the destination. However, although the intra layer interconnections can be partial, all the layers are fully connected via crossbar switches (pillar routers). Finally several forms of XNoTs topologies including meshes, torus, and/or trees are created, and they are evaluated in terms of performance, cost, and energy consumption.

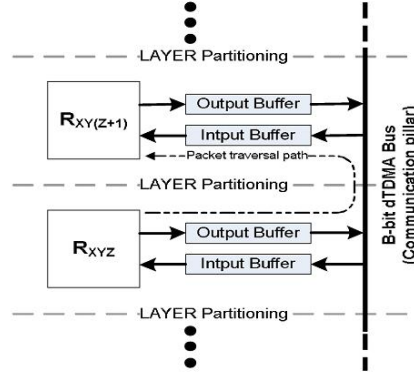


Figure 3.3: Side view of 3D NoC with the dTDMA bus[49]

In [49], Rahmani et al. propose a mechanism which benefits from a congestion-aware and bus failure tolerant routing algorithm called AdaptiveZ for vertical communication in 3D stacked mesh architecture. 3D stacked mesh is a combination of packet-switched network and bus architecture [44]. As shown in Figure 3.3, if the destination input buffer (ex: $R_{xy}(z+1)$) is full and/or dTDMA bus is busy, the packet's transmission of the source node (ex: $R_{xy}(z)$) will be blocked. Therefore as can be seen in Figure 3.4, AdaptiveZ algorithm forwards a packet from the nearest possible vertical link to the source node adaptively based on the load of vertical links (for example vertical link #1 has the highest probability; while the vertical link #7 has the lowest one). On the other hand, AdaptiveZ algorithm tries to alleviate the thermal issues by herding most of the switching activities closer to the heat sink by changing the order of intra and inter layers routing based on the location of source and destination layers: a) if the source node is located at the farthest layer from the heat sink, inter layer routing is done first and then intra layer routing; b) if the source node is located at the nearest layer to the heat sink, intra layer routing is done first and then the inter layer routing; c) if the source node is located in a layer between the furthest and nearest layers and the destination layer is an upper layer, inter layer routing will be done first otherwise intra layer routing will be done first. Finally in this way AdaptiveZ algorithm keeps the packets in source or destination layers, as nearest as possible to the heat sink. They evaluate the proposed mechanism under synthetic traffics and real benchmarks. Finally they show significant power, performance, and peak temperature improvements compared to a typical stacked mesh 3D NoC.

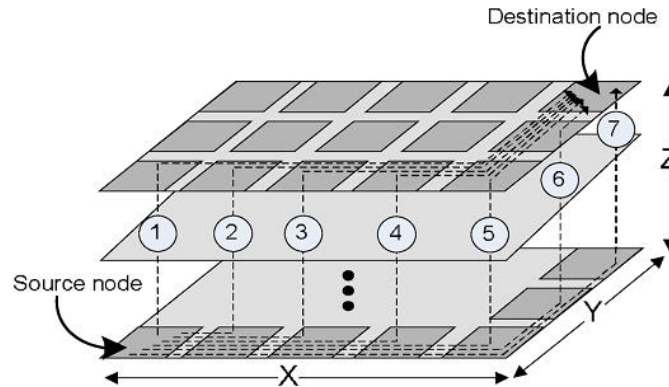


Figure 3.4: Example of 7 different possible paths in AdaptiveZ routing[49]


```

// The destination set  $\overline{D}$  is partitioned at the source tile  $s$ , separate copies of the
// packets are sent to the output channels.
for each intermediate tile  $t$ ,
    if ( $t \in \overline{D}$ )
        A packet is replicated and ejected to the local processor.
    else if  $t$  has the same  $Z$  coordinates as one of the destinations
        A packet is replicated and sent in the corresponding  $Z$  direction.
    else if  $t$  has the same  $Y$  coordinates as one of the destinations
        A packet is replicated and sent in the corresponding  $Y$  direction.
    else
         $t$  forwards the packet in the  $X$  direction.
    
```

Figure 3.5: The MXYZ routing [51]

Feng et al. [50] propose a low-overhead fault-tolerant deflection routing algorithm, which uses a layer routing table and two TSV state vectors to make efficient routing decision to avoid both vertical and horizontal link faults, for 3D NoC. The faulty links can be divided into two classes: vertical (TSV) and horizontal link. Due to the number of input ports being equal to the number of output ports in deflection switch, the faulty links are assumed to be bidirectional. For the focus on the routing algorithm, they also assume that there exists a fault diagnosis mechanism to detect faults. In order to simulate the faulty link of the switch, a 6-bit fault vector is used to represent the fault state of six links for each switch (a '1' in the fault vector represents the corresponding bidirectional link is broken). In addition, they assume that the faulty vertical links do not disconnect different layers and the faulty horizon links do not disconnect the 2D mesh for each layer to guarantee a path existing for each node pairs. In order to get the fault state of TSV, each switch contains two n-bit TSV state vectors which record the fault state of the up and down links of the current layer. The two vectors are transmitted to four neighboring switches which are in the same layer as the current switch. For each clock cycle, the switch will update its own TSV state vectors based on the TSV state vectors transmitted from its neighbors. After a short period, the switch will get all TSV state of the layer.

Wang et al. [51] deal with both regular and irregular region oriented multicasting problem 3D NoC systems. Correspondingly, two topology oriented tree based multicast routing algorithms, MXYZ is proposed for supporting multicasting in regular 3D NoC and AL+XYZ is presented for supporting multicasting in irregular 3D NoC. The MXYZ routing algorithm

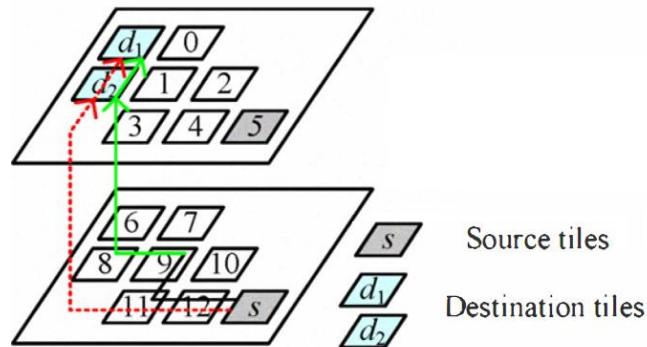


Figure 3.6: An irregular region oriented multicasting framework [51]

of MXYZ is shown in Figure 3.5. However, as can be seen the red dotted line in Figure 3.6, if the output channel found by MXYZ is not available (i.e. the tile connected to the output channel is not in the same multicast region), an alternative output channel is selected to reach the destination by AL+XYZ. In order to cope with this issue, each router has 4 connectivity bits, CNorth, CEast, Csouth, and Cwest, and each of these bits defines the connectivity at the specific output direction. Suppose a tile has coordinate (x, y, z) , and CNorth is 1 if tile (x, y, z) and its north neighbor tile $(x, y-1, z)$ are in the same region. Similarly, Cx set to 1 if the tile and its neighbor tile in the x direction are in the same region. The connectivity bits are used to find the multicasting regions in different layers. MXYZ is evaluated against a path-based regular topology oriented multicast routing and AL+XYZ against an irregular region oriented multiple unicast routing algorithm. The experimental results demonstrate that the proposed MXYZ and AL+XYZ schemes have lower latency and energy consumption than the conventional path based multicast routing and the multiple unicast routing algorithms.

Akbari et al. [52] address the low yield ratio problem in 3D NoC, due to the not-matured fabrication process of TSV connections by proposing a novel deadlock-free routing algorithm, called AFRA, for 3D mesh-based NoCs which can tolerate vertical link faults. The concerned 3D NoC architecture is fully-connected inter and intra layers. When a node realized that its vertical link is faulty it has to send its packet to a node called “escape node” which is found by the proposed routing algorithm. The proposed routing algorithm has been proved to be deadlock-free. AFRA is evaluated, though cycle accurate network simulation, and is compared with planar adaptive routing. Results reveal that AFRA improves planar adaptive routing in both synthetic and real traffic patterns.

Chen et al. [53] consider a special type of 3D NoC architecture called Non-Stationary Irregular Mesh (NSI-Mesh). To keep the temperature below a certain thermal limit, the thermal emergent routers are usually throttled. Then, the topology of 3D NoC becomes a Non-Stationary Irregular Mesh (NSI-Mesh). Therefore, they tackle the thermal issue in 3D NoC by proposing a novel Topology Aware Adaptive Routing (TAAR) to balance the traffic load in (NSI-Mesh) topology. TAAR has three routing modes (Adaptive west first, XY, and vertical balance downward routing), which can be dynamically adjusted based on the topology status of the routing path. In addition to increase routing flexibility, the TAAR also increases both vertical and lateral path diversity to balance the traffic load. Compared with the related adaptive routing methods, the experimental results show that the proposed TAAR can reduce 19% ~ 295% traffic loads in the bottom logic layer and improve around 7.7%~380% network throughput. According to the proposed VLSI architecture, the TAAR needs less than 24.8% hardware overhead compared with the previous works.

Ying et al. [54] present two generalized routing algorithms for Vertically-Partially-Connected 3D-NoC topologies, which can maintain the performance of the NoC and improve the utilization of TSV. The main contribution of this paper is a routing algorithm called Source Based Shortest Manhattan (SBSM) and Destination Based Shortest Manhattan (DBSM). The algorithm finds the nearest possible TSVs, both near the source and destination nodes in source and destination layers. Then the proposed algorithm determines in each

intermediate layer the proper path based on the position of the calculated TSVs. The experiments were done in SystemC and RTL. They compare the results of different parameters such as execution time, average throughput, system-interconnect and TSV energy consumption, and TSV utilization with the yield for 100% (fully-connected), 50%, and 25% Vertical channel Density (VD). Finally they shown that the 50% vertical channel density configurations are reasonable balance to be the best design solution, since they can maintain the system performance in comparison to 100% vertical channel density and Z-first routing while its yield ratio is 0.9.

2. 3D NoC Router Micro-Architecture Related Works

Kim et al. [62] integrate the increasingly popular idea of NoC into a 3D setting. They propose one of the initial router micro-architectures of 3D NoC. They also present a detailed exploration of the proposed inter-layer communication architecture in 3D NoCs. Three design options are investigated; 1) a simple bus-based inter-wafer connection, 2) a hop-by-hop standard 3D design, and 3) a full 3D crossbar implementation. The authors propose a novel Vertically-Partially-Connected 3D crossbar structure, called the 3D Dimensionally-Decomposed (DimDe) Router, which provides a desirable tradeoff between circuit complexity and performance benefits. They utilize virtual channels to avoid deadlock and separate the router architecture into three modules: Row (for x dimension packets), Vertical (for z dimension packets), Column (for y dimension packets). Simulation results using (a) a stand-alone cycle-accurate 3D NoC simulator running synthetic workloads, and (b) a hybrid 3D NoC/cache simulation environment running real commercial and scientific benchmarks, indicate that the proposed DimDe design provides latency and throughput improvements of over 20% on average over the other 3D architectures, while remaining within 5% of the full 3D crossbar performance. Furthermore, based on synthesized hardware implementations in 90 nm technology, the DimDe architecture improves all other designs by an average of 26% in terms of the Energy-Delay Product (EDP).

Park et al. [63] propose a 3D stacked NoC router architecture, called MIRA, which unlike the 3D routers in previous works, is stacked into multiple layers and optimized to reduce the overall area requirements and power consumption. They discuss the design detail of 3D multi-layered router (3DM) and its enhanced version with additional express channels called 3D multi-layered router with express paths (3DM-E). Finally they compare them against a (6×6) 2D design and a baseline 3D router (3DB) design. All the designs are evaluated using a cycle-accurate 3D NoC simulator, and integrated with the Orion power model for performance and power analysis. The simulation results with synthetic and application traces demonstrate that the proposed multi-layered NoC routers (the 3DM and 3DM-E) can outperform the 2D and naive 3D designs in terms of performance and power. It can achieve up to 42% reduction in power consumption and up to 51% improvement in average latency with synthetic workloads. With real workloads, these benefits are around 67% and 38%, respectively.

Lafi et al. [64] present a new router that enables gains in terms of throughput and latency compared to classic 3D mesh the in case of large NoCs. The proposed router is hierarchical

since it is composed of two totally decoupled modules: one for inter-layer communication and one for intra-layer communication. Throughput and latency evaluation is performed using a SystemC-TLM NoC simulator. Synthesis and extrapolation results show that the hierarchical router is competitive with the classic 3D mesh in terms of area and power. Simulations' results show that the proposed hierarchical router can improve 3D mesh by more than 30% in terms of throughput and latency in the case of transpose traffic. However, Lafi et al. [64] did not accurately implement the proposed router to calculate area and power consumption they assume a vertically fully-connected 3D NoC-based SoC.

Rahmani et al. [49] proposed a 3D router architecture for AdaptiveZ routing algorithm which is a congestion-aware and a bus failure tolerant algorithm. They show the area overhead of the control module of their router is trivial and the performance of their router is better compared to symmetric NoC 3D-Mesh and Hybrid NoC-bus based 3D-Mesh. Also, Darve et al. [65] propose a hierarchical 3D router. The router is composed of two coupled routers: one is for inter- and the other one is for intra-layer communication. They show that the area of the proposed router with serialized vertical links is 43% less than the one with parallel vertical links. Feng et al. [50] proposed a fault tolerant routing algorithm and its router. The proposed router is implemented in hardware with TSMC 65nm technology, which can achieve 250MHz. Compared with a reinforcement-learning-based fault-tolerant deflection router with a global routing table; the proposed switch occupies 40% less area and consumes 49% less power consumption. Simulation results demonstrate that the proposed switch has 5% less average packet latency than the switch with the global routing table under real application workloads and with only 5% performance degradation under synthetic workloads in the presence of 10% link faults.

Latif et al. [66] present the partial virtual-channel sharing (PVS) NoC architecture which reduces the impact of fault on system performance and can also tolerate the faults on routing logic. In the proposed architecture autonomic virtual-channel buffer sharing is implemented. The runtime allocation of the buffers depends on incoming load and fault occurrence. This technique can be used in any NoC topology and for both 2D and 3D NoCs. The synthesis results for an integrated video conference application demonstrate significant reduction in average packet latency compared to existing VC-based NoC architecture. Extensive quantitative simulation results for synthetic benchmarks are also carried out. Furthermore, the simulation results reveal that the PVS architecture improves the performance significantly under fault free conditions compared to other VC architectures.

Rahmani et al. [67] propose a novel hybridization scheme for inter-layer communication using efficient 5-ports routers to enhance the overall system power, performance, and area characteristics of the existing Hybrid NoC-Bus 3D mesh architecture. By proposing a routing algorithm called LastZ, the proposed area-efficient architecture decreases the overall average hop count of a NoC-based system compared to the existing architectures. The authors further improve this design by proposing partial-LastZ-based 3D NoC-bus hybrid architecture to provide adaptivity for implementing congestion-aware and fault-tolerant inter-layer routing algorithms. The experiments demonstrate up to 16% performance improvement compared to

the full LastZ-based 3D NoC-bus hybrid architecture and around 20% area reduction compared to the typical hybrid NoC-Bus 3D mesh architecture.

3. 3D NoC Architectural Exploration Related Works

As NoCs are going 3D, new problems raised as compared to flat topologies. This becomes more challenging when some TSVs have been removed to improve yield ratio or to use the resources (surface) in more efficient way. In following some of the most referred and related works to architectural exploration of 3D NoC especially, vertically partially-connected ones, will be discussed.

In Bartzas et al. [55] (expanded in the First Chapter of Gebali et al. [56]), address the issue of reducing the number of vertical links in 3D Mesh-based NoCs following specific patterns to adapt the topology to the application communications flows by a fully software-supported exploration methodology. Their exploration methodology is able to evaluate pattern-based 3D topologies and propose the ones that meet the design constraints best. Unfortunately they do not explain in detail the routing algorithm based on the utilization of the temporary destinations in the intermediate layers. Additionally, each router has a routing table, which based on the source/destination addresses decides each incoming packet's output port. They evaluate the exploration employing, extending the Worm-Sim NoC simulator, and feeding it with various types of traffic (uniform, transpose, and hot spot). Their primary cost function is the energy consumption, with the other cost factors being the average packet latency and total switch block area. The comparison basis are the fully vertically interconnected 3D Mesh-based NoCs of 64-node ($4 \times 4 \times 4$) and of 144-node ($6 \times 6 \times 4$) as well the 2D Mesh-based NoCs of 64-node (8×8) and of 144-node (12×12).

Finally they show that the 2D Mesh-based NoCs have more energy consumption, greater average packet latency, and less total switch block area consumption than vertically fully-connected 3D Mesh-based NoCs. Also they show the comparisons between vertically fully and partially-connected 3D Mesh based NoC, reproduced in Table 3.1. The three types of traffic are shown in the first column. The next three columns present the gains (min. values to max. values – in %) for energy dissipation and area occupation (second and fourth column respectively). In the third column, the increase in latency is presented. As can be seen in Table 3.1, by employing 3D NoC with less vertical links (44% of node are 2D routers), they can achieve gains in energy consumption (up to 29%), in the area occupied by routers of NoC (up to 18%) and they have growth in the average packet latency (up to 2%) “in best cases”. They achieve a decrease in the number of 3D routers and in the number of TSVs, resulting in a decrease in the area occupied by the switch blocks, reducing energy dissipation and paying a reasonable penalty in the latency of the 3D NoC.

Weldezion et al. [57] examine the performance and scalability of different communication topologies for 3D NoC (2D Mesh NoC, vertically fully-connected bus-based and non bus-based 3D Mesh-based NoC) with large number of nodes (more than 1000). Cycle accurate RTL-level simulations are conducted for two communication schemes based on 1) a 7-port switch and 2) a centrally arbitrated vertical bus using two types of traffic patterns 1) uniform and 2) localized. The scalability of the 3D NoC is examined under both

Table 3.1 min-max impact on the energy, latency and area values of vertically partially 3D NoC that were obtained are compared to the ones of the 3D full vertically interconnected NoC in [55] and [56]

64-node architecture	Energy	Latency	Area
Uniform	98.9% – 94.6%	102.9% – 114.3%	92.9% – 82.2%
Transpose	98.9% – 92.9%	102.2% – 108.4%	94.7% – 92.9%
Hotspot	96.9% – 87.5%	102.8% – 117.6%	94.7% – 82.2%
144-node architecture	Energy	Latency	Area
Uniform	97.8% – 92.4%	101.3% – 106.7%	94.7% – 82.2%
Transpose	97.2% – 92.7%	102.1% – 112.3%	94.6% – 87.5%
Hotspot	99% – 97.5%	103.3% – 103.7%	94.7% – 92.9%

non bus-based and bus-based 3D Mesh-based NoC architectures and compared to 2D NoC structures in terms of throughput and latency. Finally they showed that bus-based 3D Mesh-based NoC has reasonable latency and performance for smaller than (3×3×3) 3D NoC, low injection rates (0.1 per cycle) localized traffic. While 3-D 7-port switch behaves reasonably for high injection rates (0.2 and 0.3 per cycle). They show that the bus-based 3D NoC has advantages in other traffic conditions, such as when a stack of memory lies above a processor, where the bus will provide equal access time to any layer of memory. Finally they conclude that under uniform random patterns, the 7-port switch is the best option for large sized networks in term of packet throughput, but trade offs such as area overhead and power may render the bus a viable alternative, especially when the vertical links are limited, and resources are scarce.

Another most related work to this thesis, is Xu et al. [59]. Due to the criticality of finding a balance between performance and manufacturing cost in 3D NoC, Xu et al. [59] model a 3D NoC, study the placement of TSVs and the cost of TSVs, and analyze the performance with different number of TSV using a full system simulator. As a case study to find the optimal placement of TSVs, they chose an 8×8 Mesh layer. The number of possible placements of 16 vertical links in an 8×8 mesh is too large. Therefore, they explore the possibilities starting from a smaller NoC by writing a program that exhaustively enumerate all placement possibilities and output the combinations of lowest Average Hop-Count. They use divide and conquer methodology to deal with the large exploration space. They divide an 8×8 Mesh layer with 16 vertical links into four 4×4 Mesh layers, each with 4 vertical links. Finally by analyzing of the results of possible placements (by exhaustive simulation) in an 4×4 Mesh layer they found two patterns which two are such that vertical links are exactly three hops far from each other and in each row and column there are one vertical link. They try to place the patterns on an 8×8 Mesh layer with different positions to find the optimal vertical links placement. Their experimental results have shown that comparing with 8 vertical links (25% of nodes are 2D routers) in 8×8 Mesh layer, the average network latencies in 8×8 Mesh layer with 64 vertical links (vertically fully-connected) and 16 vertical links (50% of nodes are 2D routers) are reduced 14.7% and 7.38% respectively.

CHAPTER THREE: STATE OF THE ART

They assume that the number and placement of the TSVs are the same in each layer. Also, they propose a fixed regular pattern for the placement of the TSVs in which one 3D node is attached to three 2D nodes just by one hop-count. This might be a bottleneck based on the transactions of the attached nodes. Also NoCs with irregular sizes and irregular number of vertical links, e.g. a 13×13 mesh with 53 pillars, are not suitable for their method which is a considerable limitation. However, their idea can be expanded to NoCs with regular sizes and regular number of pillars, e.g. an 8×8 mesh with 4 vertical links (divided into 4 meshes, each 4×4 with 1 vertical link), or a 9×9 mesh with 18 vertical links (divided into 9 meshes, each 3×3 with 2 vertical links).

Chen et al. [60] also evaluate a vertically partially-connected 3D FPGA-based NoC. They introduce two methods to reduce area consumption: a) less number of TSVs in each 3D router and b) less number of 3D routers through each layer. In addition, they evaluate a hybrid architecture based on these two methods. However, they assume that 1) all the tiers are the same size, 2) all the 3D routers, in the first method, should have the same number of TSVs in all the layers, and 3) the number of 3D routers, in the second method, must be the same in all the layers. Therefore, although this work is interesting, more flexibility is required to meet the needs for a complete evaluation of the different architectural possibilities and behaviors. Finally, the authors recommend several configurations for generic 3D FPGA architectures, which can save up to 52% area.

Table 3.2: Summary of the presented related works; NA: Not Applicable

Reference	Category	System				μ -Architecture	
		Interconnection		Routing Deadlock freedom is a must!		Router	
		Vertically Fully	Vertically Partially	Deterministic	Adaptive	Bus-based dTDMA bus	Non Bus-based
Li <i>et al.</i> [42]	3D NoC Integration		◆	◆		◆	
Pavlidis <i>et al.</i> [43]		◆		◆		◆	
Feero <i>et al.</i> [44]		◆		NA		◆	◆
Seiculescu <i>et al.</i> [45]		◆		NA			◆
Matsutani <i>et al.</i> [48]		◆			◆	◆	
Rahmani <i>et al.</i> [49]	Chapter Four 3D NoC Routing Algorithm	◆			◆	◆	
Feng <i>et al.</i> [50]		◆			◆		◆
Wang <i>et al.</i> [51]		◆			◆		◆
Akbari <i>et al.</i> [52]		◆			◆		◆
Chen <i>et al.</i> [53]			◆		◆		◆
Ying <i>et al.</i> [54]			◆	◆			◆
Kim <i>et al.</i> [62]			◆	◆		◆	
Park <i>et al.</i> [63]	Chapter Five 3D NoC Router μ -architecture	◆	◆	◆		◆	
Lafi <i>et al.</i> [64]		◆		◆			◆
Darve <i>et al.</i> [65]		◆		◆			◆
Latif <i>et al.</i> [66]		◆			◆		◆
Rahmani <i>et al.</i> [67]		◆			◆	◆	
Bartzas <i>et al.</i> [55]			◆	◆			◆
Gebali <i>et al.</i> [56]	Chapter Six 3D NoC Architectural Exploration		◆	◆			◆
Weldezion <i>et al.</i> [57]		◆		◆		◆	◆
Xu <i>et al.</i> [59]			◆	◆			◆
Chen <i>et al.</i> [60]		◆		NA			◆

4. Conclusion

In this chapter the previous works which are related to the domain of this thesis have been presented. First of all, some studies based on the concepts of 3D NoC based SoC have been explained. Then particularly the related works have been discussed based on the three main issues of this thesis respectively which are summarized and categorized in Table 3.2 based on the following domains:

- **The routing algorithms of 3D NoC**, which will be discussed in Chapter Four, solve the routing problem of Vertically-Partially-Connected 3D-NoC.
- **The micro-architecture of routers in 3D NoC**, which will be explained in Chapter Five, design router architectures to improve the performance of Vertically-Partially-Connected 3D-NoC from micro-architectural point of view.
- **The architectural explorations of 3D NoC**, which will be studied in Chapter Six due to improve the performance of Vertically-Partially-Connected 3D-NoC from system level and architectural exploration point of view.

CHAPTER FOUR: ELEVATOR FIRST ROUTING ALGORITHM

As described before, in Vertically-Partially-Connected 3D-NoC some vertical links are removed to reach higher yield and better usage of silicon area for logic circuits and interconnection. In such architectures the number, placement, and data flow direction of vertical links, inter different layers, are determined based on design constraints and might be different from layer to layer, as well as the topology of the network in each layer. However, the Vertically-Partially-Connected 3D-NoC architectures on which this thesis focuses contain stacked mesh-based layers where every layer can be different in technology and size (the number and size of tiles). The main problem of Vertically-Partially-Connected 3D-NoC is routing due to the removed vertical links.

In this chapter a deadlock- and livelock-free routing algorithm is presented to address the disconnectivity problem in Vertically-Partially-Connected 3D-NoC. Since routing algorithm has a key role in the performance of Network-on-Chip, its efficiency is important. Also, the routing simplicity makes the router micro architecture more manufacturable. Consequently the functionality of routing algorithm affects the average latency of packets and power and area consumption of links and NoC routers. Moreover, the deadlock-freedom of a routing algorithm is a necessity which definitely has to be considered. In order to this, in the following, the concept of Elevator and Elevator First routing algorithm are demonstrated. Afterwards the deadlock issue in Elevator First routing algorithm is discussed. Finally the performance of the routing algorithm is shown.

1. Elevator Concept

Fundamentally in Vertically-Partially-Connected 3D-NoC, when a packet is issued, if the source and destination nodes are in the same layer, one of the conventional deadlock-free routing algorithms can be utilized. As mentioned before, in this thesis mesh-based topologies have been selected as the topology of each layer. Also, “X-First” routing algorithm has been selected to use for horizontal intra layers transactions. Figure 4.1 depicts the X-First routing algorithm in a 2D mesh NoC. As can be seen, node 00 (source node) wants to send a packet to node 21 (destination node). X-First routing algorithm is a dimension order routing which

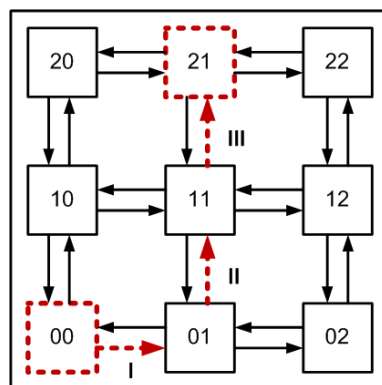


Figure 4.1: X-First Routing Algorithm as intra layer routing mechanism

routes packets first in X-direction to the correct column and then in Y-direction to the destination. Suited to the networks using mesh or torus topologies, X-First routing algorithm is well. Addresses of the routers are their XY-coordinates. This dimensional order makes the X-First routing algorithm deadlock-free [69]. This deadlock freeness and simplicity of X-First routing are the reasons that it has been chosen in this thesis to transmit the intra layer packets.

On the other hand in Vertically-Partially-Connected 3D-NoC, if the source and destination nodes of a packet are in different layers, algorithmic dimension-ordered routing mechanism, such as Z-First, cannot be utilized as the network is potentially irregular. The problem occurs when a node, which does not have any vertical links, wants to send a packet to upper or lower layer. In such case the node can use an intermediate node in the same layer called Elevator (3D route) which is connected to UP and/or DOWN port(s). Consequently the main idea behind the Elevator First routing algorithm is utilizing intermediate nodes (Elevators) that have vertical links to send the packets of nodes that do not have ones. Elevator has three types, since in Vertically-Partially-Connected 3D-NoC some routers (nodes) do not have any vertical links (UP or/and DOWN ports). In Figure 4.2, the types of Elevators are presented. If a router has only an UP port and is selected to forward packets to upper layers is called UP-Elevator (Elv_u) (see Figure 4.2.a). Also, if a router has only a DOWN port and is selected to forward packets to lower layers is called DOWN-Elevator (Elv_d) (see Figure 4.2.b). If a router has both of UP and DOWN output ports and is selected to forward packets to both upper and lower layers, it is called an UP/DOWN-Elevator (Elv_u/d) (see Figure 4.2.c).

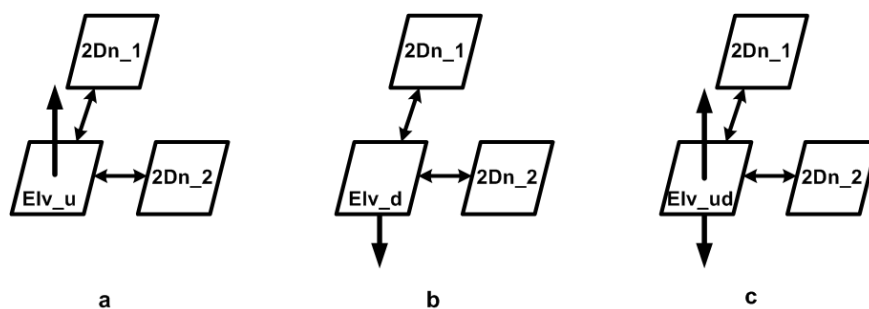


Figure 4.2: Elevator types

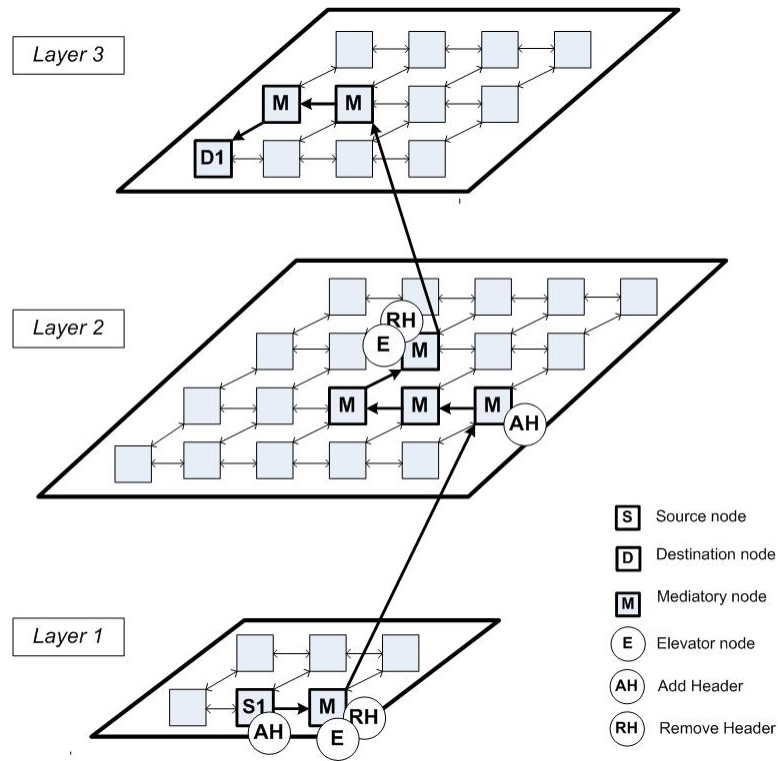


Figure 4.3: Elevator First Routing Algorithm

2. Elevator First Routing Algorithm

In the previous section the concept of Elevator has been discussed. In this section the procedure of decision making and packet routing in Elevator First routing algorithm are presented. For using Elevator First routing algorithm, every node in Vertically-Partially-Connected 3D-NoC should be attributed to two addresses: a) address of an Elv_u or Elv_u/d and b) address of an Elv_d or Elv_u/d. Therefore, when a router (node) has a packet which has to be forwarded to an upper or lower layer, this router directly forwards the packet to its UP Elevator or DOWN Elevator. Figure 4.3 illustrates an example of the Elevator-First routing. In layer 1, node S1 issues a packet whose destination (D1) is in the upper layer (layer 3) and S1 does not have a vertical link to this layer. Therefore, S1 should forward the packet to its Elv_u which is in the same layer (Layer 1). In order to do this, S1 adds a temporary header to the packet which contains 1) the address of its Elv-u and 2) two flags:

- Flag_T: If T flag is '1', it means that current header is a temporary header. While if the T flag is '0', it means that current header is a permanent and the main header of the packet.
- Flag_U: If U flag is '1' it means that this packet should be forwarded to the upper layer. While if U flag is '0' it means that this packet should be forwarded to the lower layer.

Finally, S1 routes the packet with its temporary header to the Elv_u by using the X-First routing mechanism. When the Elevator receives the temporary header by checking the T flag, it realizes that this header is temporary and by checking the U flag it realizes that the real destination is in the upper layer. The temporary header is then removed and the packet simply

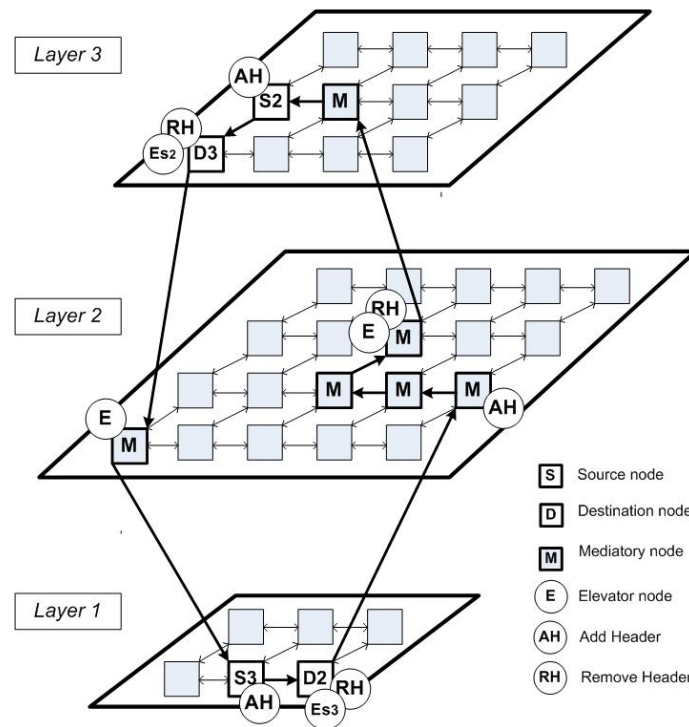


Figure 4.4: Deadlock in Elevator First Routing Algorithm

forwarded to the UP output port. After the packet has been received by the DOWN input port of mediatory node in the upper layer (Layer 3), it should be ascertained whether or not the current layer is the destination layer. If not, the header addition procedure should be done again in order to send the packet to Elv_u of the current node, and so on. Otherwise, if this layer is the destination layer, the original header will be kept and the X-First routing will be utilized to route the packet to its destination (D1) (see Figure 4.3).

2.1. Deadlock in Elevator First Routing Algorithm

Unlike Z-First routing algorithm as a dimension order algorithm is deadlock-free, originally Elevator-First routing algorithm is not deadlock-free. A case of dead-lock is presented, in Figure 4.4. Imagine that S3 wants to send its packets to D3 and at the same time S2 wants to send its packets to D2. It can be seen in Figure 4.4 that the connections between (S2 and D3) and (S3 and D2) have been shared by both routing paths from S2 to D2 and from S3 to D3. Since the routing is wormhole with limited capacity of buffers, deadlock happens.

3. Solution for Deadlock in Elevator First Routing Algorithm

To cope with deadlock problem in Elevator First routing algorithm, the channels of the 2D plane must be able to separate the packets where destinations are in upper layers from the packets where destinations are in lower layers (see Figure 4.5). We therefore introduce to virtual channels for each physical channel in the 2D plane and call them Z+ and Z-. The Z+ virtual network is composed of 2D Z+ virtual channel ports and the ascending vertical channels which contain the UP output port and DOWN input port. Similarly, the Z- virtual

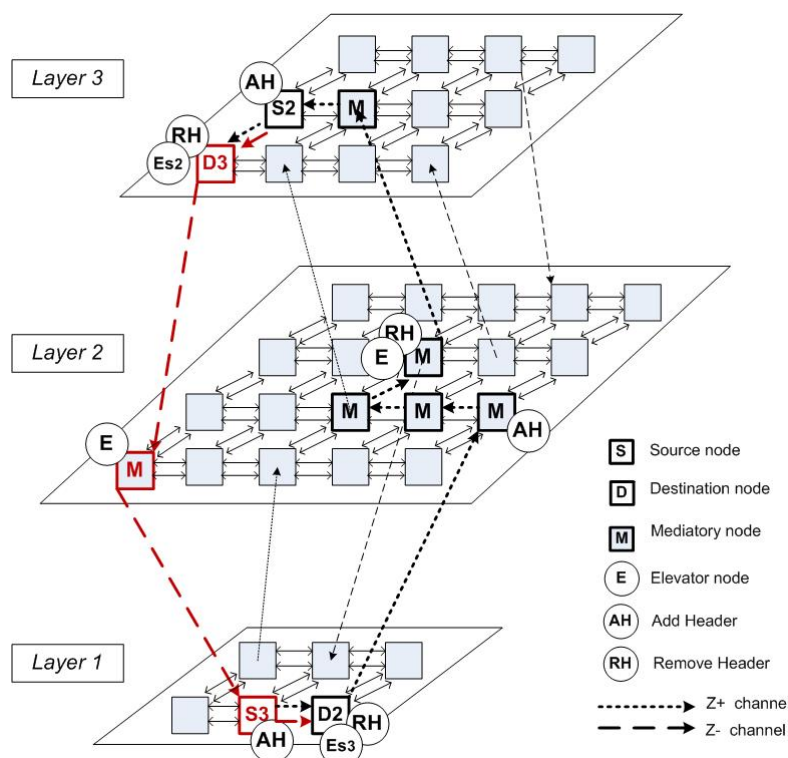


Figure 4.5: Elevator First Routing Algorithm with Z+ and Z- Virtual Networks

network contain the UP input port and the DOWN output port. The Z+ and Z- virtual networks are entirely independent from each other and the packets of one virtual network never pass through the other one. The formal proof of the Elevator-First routing deadlock freedom is presented in [68].

4. Elevator First Routing Algorithm with Z+ and Z- Virtual Networks

Fundamentally, in a mesh-based architecture, addresses are composed of a triple of (X, Y, Z). Elevator-First routing modules deals with four types of address to decide a proper output port: a) Current Router Address (ADRC), b) Packet Destination Address (ADRd), c) Elv_u Address (ADRez+), and d) Elv_d Address (ADRez-). The addresses of both Elv_u and Elv_d are initially assigned to every node and are registered in two registers. Note that even if a router has an UP or/and a DOWN output port, the addresses of its Elv_u and Elv_d can be different from its own address, for example, because of the traffic distribution that the designer may desire.

The block diagram of the Elevator First router with 2D ports (N, W, S, E), 3D ports (U, D) and LOCAL port is shown in Figure 4.6. Every input port has a routing module which performs Elevator First routing algorithm. The Elevator-First routing algorithm in the local input port is presented Figure 4.7. Firstly, when the routing module receives a header, it compares the z part of the destination address (ADRd) of the packet, z_d , with z part of the current router address (ADRC), z_c . If z_d is the same as z_c (i.e. the destination and source nodes are in the same layer), a round-robin finite state machine (RR FSM) decides which of Z+ or Z- virtual networks should be selected. The round-robin strategy, by which if the previous packet has been forwarded for example to the z+ network the current packet should be forwarded to the z- network, aims to balance the load between the two virtual networks.

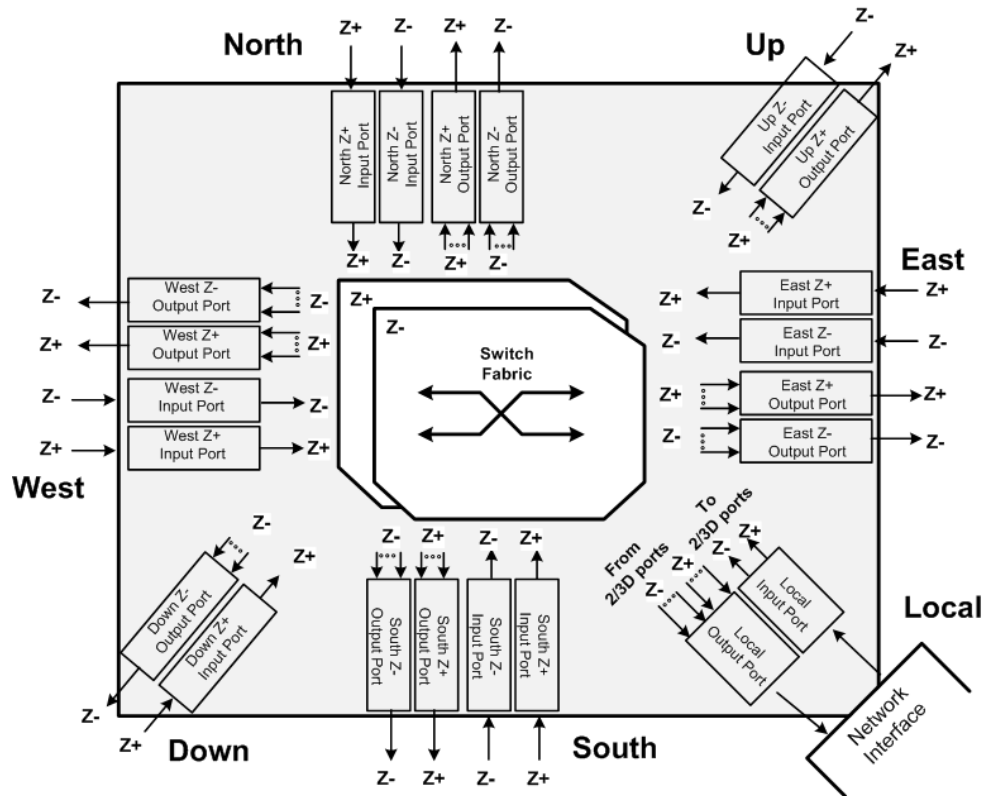


Figure 4.6: The block diagram of Elevator First Router

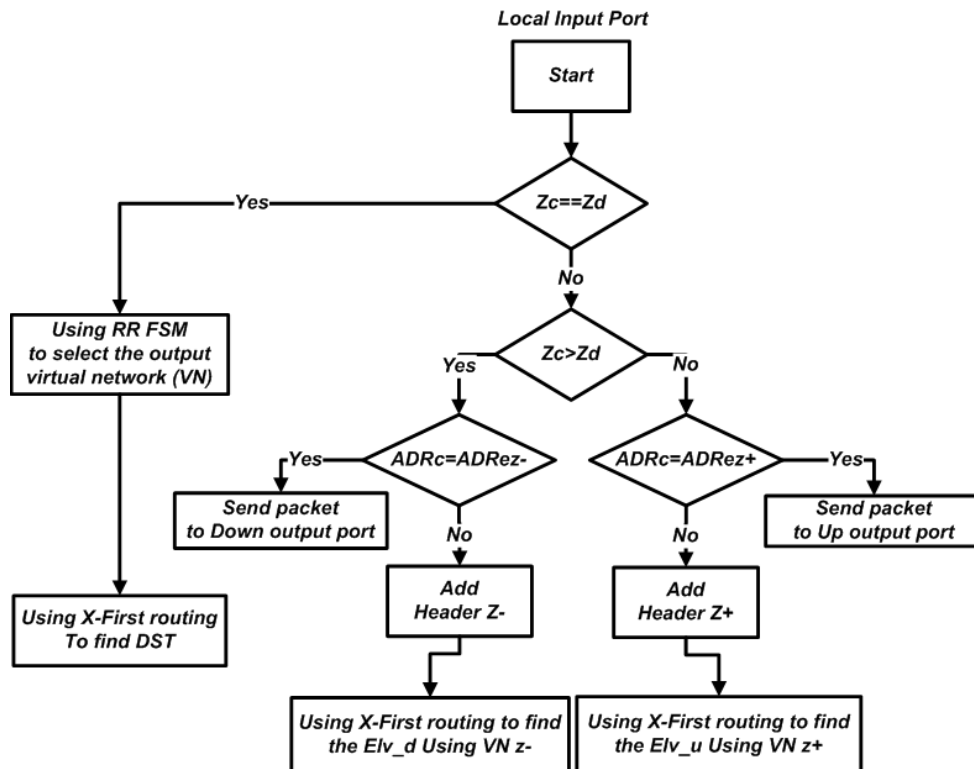


Figure 4.7: Elevator First Routing in LOCAL input port

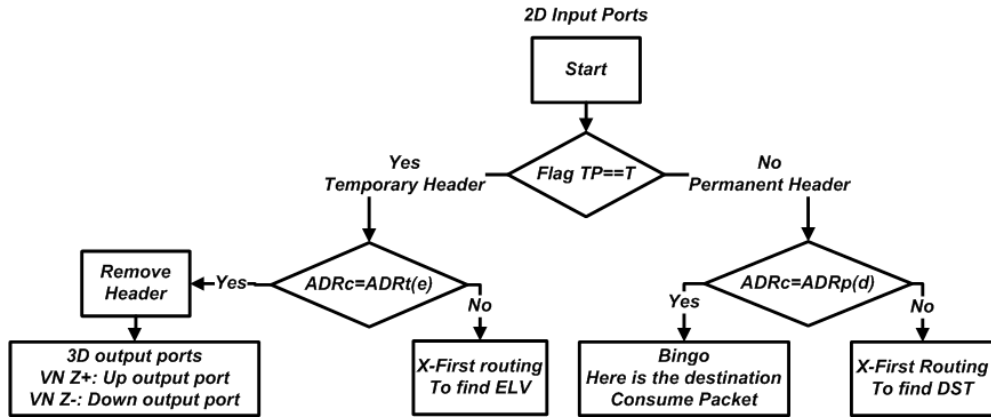


Figure 4.8: Elevator First Routing in 2D input ports

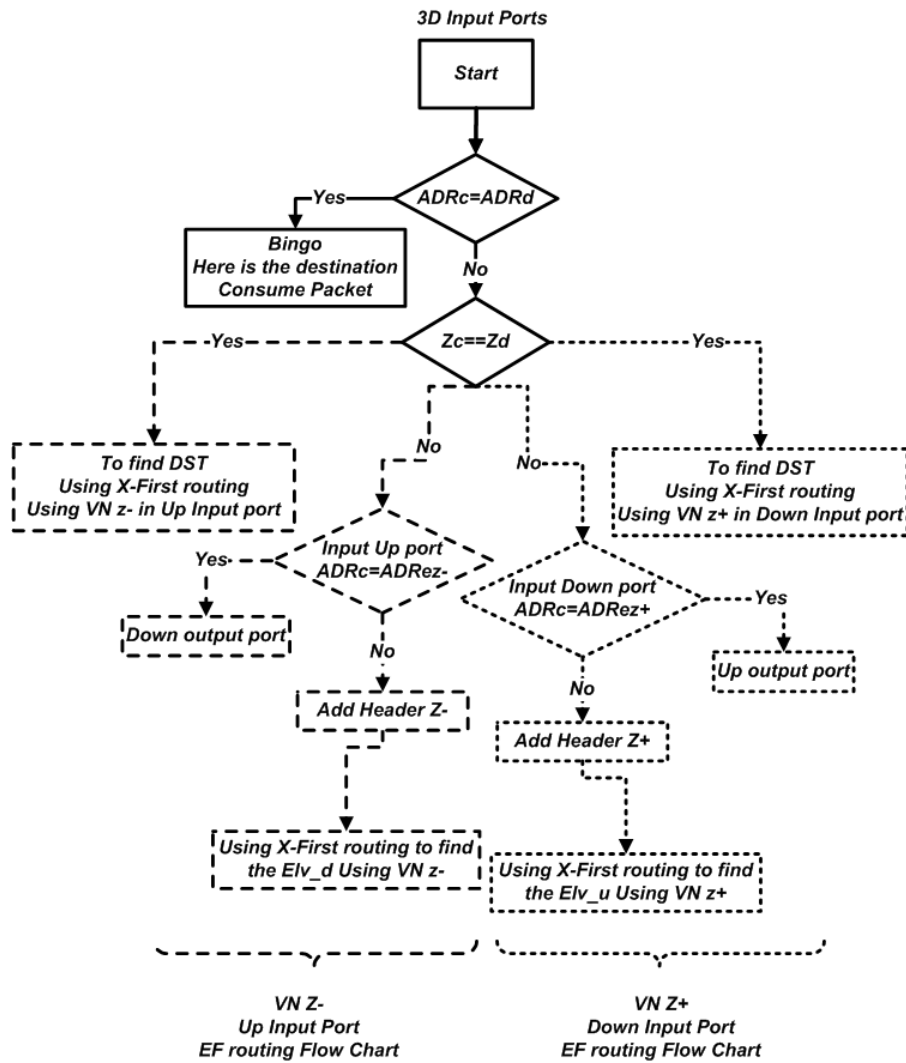


Figure 4.9: Elevator First Routing in 3D input ports

On the other hand if Z_d is not the same value as Z_c , LOCAL routing module compares $ADRC$ with the corresponding Elevator addresses ($ADRez^+$ or $ADRez^-$). If the current node is its own Elevator, which means the current router has the desired vertical link, the corresponding 3D output port will be selected to forward the packet. Otherwise, the LOCAL

input port adds a temporary packet header with Elv_u or Elv_d address (with regard to the destination layer) as a temporary and intermediate destination and forwards the packet to the corresponding 2D output port (determined by X-First routing algorithm) of the corresponding virtual network (Z+ or Z-).

In a same manner, we describe the Elevator-First routing algorithm in 2D and 3D input ports in Figure 4.8 and Figure 4.9 respectively. While only the arbitration between concurrent packets targeting a same output is done in output ports, all routing decisions are made in input ports to send packets to the appropriate output port. Adding temporary headers is done in 3D input ports like in LOCAL input port. Likewise, the removal of temporary headers is carried out in 2D input ports when the packet should go up or down. When 2D input ports receive a packet, first they will check if the packet header is temporary or permanent (by means of flag T). If the header is temporary and current node is the destination, it means that the current router is the Elevator. Therefore, 2D input port removes the temporary header and forwards the packet to the corresponding 3D ports based on the value of flag U. Otherwise, if the header is permanent, it means that current layer is the destination layer and consequently the 2D input port utilizes X-First routing algorithm to find the output port toward the destination.

5. Experimental Results of Elevator First Routing Algorithm

In order to study the practicality of the Elevator First routing algorithm a Cycle-Accurate SystemC model of Elevator First router has been developed. Figure 4.10 and 4.11 show the average latencies of a $5 \times 5 \times 5$ network versus the flit injection rate (offered load of cores, as the fraction of maximum possible load, i.e. one flit per cycle) in different situations when respectively using uniform traffic distribution (in which each core sends packets uniformly to all other cores with a same probability and a same rate) and localized (when closest destinations of being reached have a higher probability).

For these simulations routers with 16-flit deep FIFOs and 16-flit packet size have been used. The curve “UniNorm” presents the average latency of a fully connected 3D-Mesh network when using the ordinary Z-First routing algorithm. We firstly analyze Figure 4.10. The curve “Uni0” presents the average latency using the Elevator First routing algorithm when 0% of vertical links are removed. It means the network is a fully-connected 3D-Mesh, but it uses two virtual channels (buffers) of Z+ and Z- for ascending and descending packets respectively, and the (UP and DOWN) Elevator of each node is itself. As can be observed, separating the ascending and descending packets into two independent Z+ and Z- networks has a positive influence on the network performance, as the traffic and consequently the contention probability is divided by 2. To make the comparison a little bit fairer, the ordinary network with larger FIFOs have been simulated to have the same overall amount of resource as “Uni0” with 2 VCs. The curve “UniNormLF” presents this case and as can be seen the network performance is better than the case of “UniNorm”, but there exists yet a gap between the performance of “Uni0” and “UniNormLF” even if the buffering resources for the two cases are the same. The curves “Uni5”, “Uni10”, “Uni25”, and “Uni50” are the results when 5%, 10%, 25%, and 50% of vertical links are removed respectively.

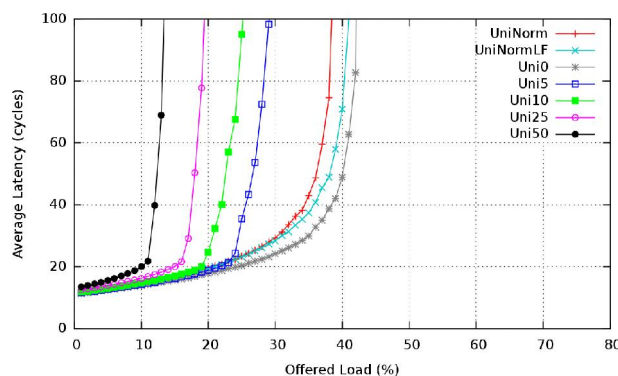


Figure 4.10: Average latency using uniform traffic distribution

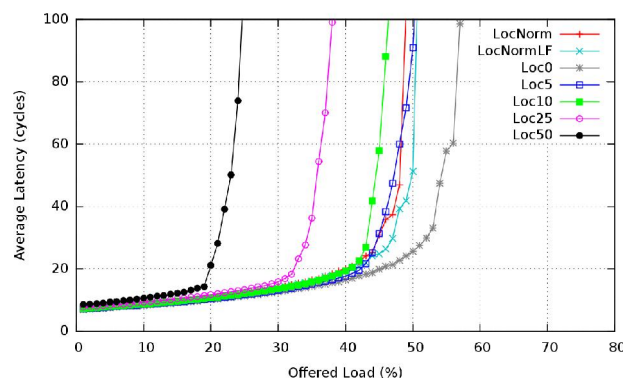


Figure 4.11: Average latency using localized traffic distribution

In our simulation platform, the choice of which vertical links to removed is chosen randomly, and in order to have a more uniform distribution of missing vertical links the simulations have been repeated 20 times. Note that we did not impose any constraints on the selection of vertical links to be removed and in the modelled topologies there might be routers without an ascending link (UP PORT), descending link (DOWN port), or either. The only point has to be taken into consideration is that in each layer there must be at least one ascending and one descending channel in order to keep the network connected. For nodes with removed vertical link(s), one (two) Elevator(s) is (are) selected randomly between the nearest routers in which the corresponding vertical link is not removed.

Figure 4.11 demonstrates the simulation results similar to those of Figure 4.10, but this time for localized traffic distribution [57]. In the localized traffic distribution the probability of sending a packet to a destination decreases exponentially with the destination distance. In other words, with a localized traffic distribution, cores send more packets to nearer nodes and less to farther ones. As can be seen in the Figure 4.11, even when 10% of vertical links are removed the network performance is approximately the same as that of a fully connected 3D-Mesh network using the usual Z-First routing algorithm (i.e. “LocNorm”).

5.1. Average Hop-Count

Elevator First routing algorithm is not minimalist and the path between source and destination may be longer than the Manhattan distance. The average hop-count that shows the average number of routers a packet should pass to reach its destination could be a good parameter to describe how Elevator First routing degrades the path length. The hop counts not only gives a fair approximation of packet latencies in low traffic loads, but also have a direct influence on the energy consumption of packets to arrive to the destinations. Assuming average power consumption of each router is P and the average hop counts of packets is n , it can be roughly estimated that the power consumed by the network is proportional to $n \times P$. Of course this is a rough estimation and to measure the real power consumption there are many other parameters that should be taken into consideration. The average hop-counts of the aforementioned topologies for the two types of uniform and localized traffic distributions are shown in Figure 4.12 and Figure 4.13 respectively. In these experiences one of the nearest 3D

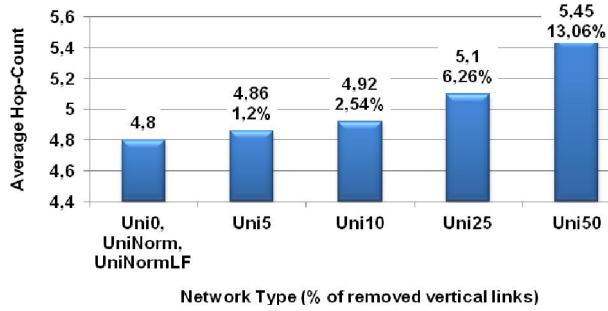


Figure 4.12: Average Hop-Count using uniform traffic

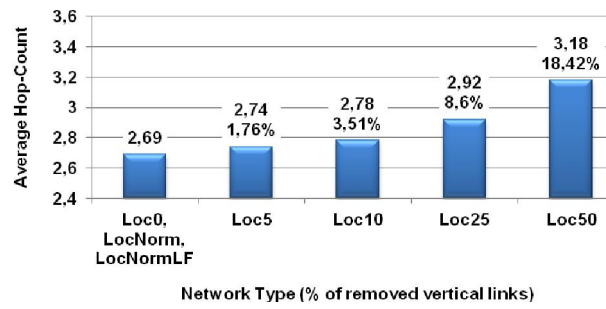


Figure 4.13: Average Hop-Count using localized traffic

routers is attributed as the elevator of routers, and indeed by changing this strategy the average hop-count changes. As can be seen, by increasing the number of removed vertical links the average hop-count increases. And, although the average hop-count in the case of localized traffic is less than in the uniform traffic case, Elevator First routing degrades the average Hop-Count of the localized traffic case proportionally more than in the case of uniform traffic. However even with 50% missing vertical links in localized traffic distribution, the average hop-count degradation is about 18%. In most situations, such a degradation seems to be acceptable.

6. Conclusion

In this chapter, to cope with the routing problem in Vertically-Partially-Connected 3D-NoC architectures, the Elevator First routing algorithm as a deadlock- livelock-free efficient solution has been presented and analyzed. The combination of light and simple Elevator First routing algorithm and the separation of ascending and descending packets by two Z+ and Z-virtual networks, not only makes the Vertically-Partially-Connected 3D-NoC architecture completely connected, but also as it could be seen in results section, provides a practical and low latency infrastructure for the transactions.

CHAPTER FIVE: MICRO-ARCHITECTURE OF ELEVATOR FIRST ROUTER

The NoC router, as shown in Figure 5.1, is the hardware infrastructure for performing the packet routing. Therefore, any inefficiency in the design of router has a significant impact on the performance of the system. In addition, the router design affects the area and power consumption of NoC architecture which are the key cost design parameters. In this chapter, firstly the DSPIN router (see Figure 5.1) is presented and then the design and implementation of the proposed router's micro architecture for Vertically-Partially-Connected Mesh-based 3D-NoC based on Elevator First routing algorithm are described.

1. DSPIN Router

Here the micro architecture of the five ports “DSPIN router” [73], which has been designed and implemented for 2D DSPIN NoC in LIP6 laboratory and developed for 3D NoC in this thesis, is presented (see Figure 5.1). As can be seen, the 2D DSPIN router has five

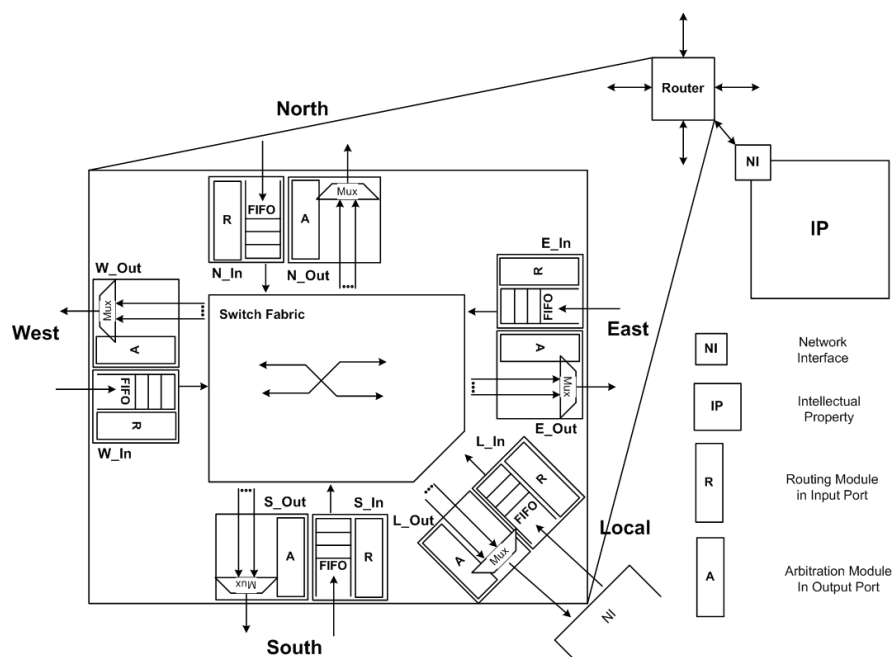


Figure 5.1: DSPIN router [73]

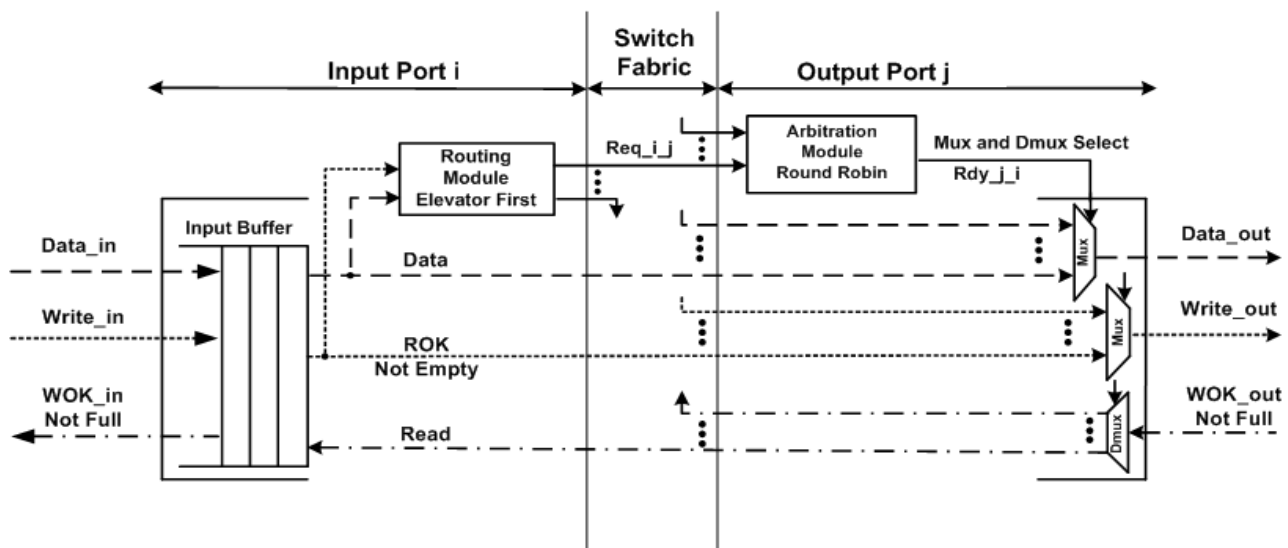


Figure 5.2: Ports handshaking in conventional NoC router

(input and output) ports, West, North, East, South, and Local. The routing and arbitration are done in a distributed way in each input and output ports respectively. The input port is responsible to receive the flits of a packet and store them in a FIFO. Also, the routing module (R) in input port performs the routing algorithm to determine the destination port. On the other hand, in output port, the arbitration module (A) decides which input port, among those which have already (simultaneously) a packet to send, has to be served. The arbitration chooses the proper input port by issuing the select signal of the output multiplexer. Switch Fabric is the internal connections of the router between the ports.

In DSPIN router, as can be seen in Figure 5.2, when an output port wants to read from the buffer located in the input port a) the Read signal (which is used as the read command of the buffer) must be enabled; b) the ROK signal (standing for “Read OK” which represents that the buffer is not empty so there is a valid data to be transferred) must be enabled. Identically, when an output port wants to write in the input buffer of the downstream router, a) the Write signal (which is used as the write command of the buffer) must be enabled, b) the WOK signal (standing for “Write OK” which represents that the buffer of the downstream router is not full) must be enabled. We use ROK of the current buffer as Write for the downstream router, and WOK of the downstream router as Read for the current buffer. Finally if the downstream router has enough space in its input FIFO, the WOK signal becomes enable, which allows the flits in input port ‘i’ to be written in the downstream input port FIFO. Technically, the flits are written when both “ROK” and ”WOK” signals are ‘1’ which mean there is a flit in input port ‘i’ and there is enough space in downstream input port.

In addition, in DSPIN NoC in each flit of a packet, there are two reserved bits (flags) called “BoP” which indicates begin of packet, and “EoP”, which indicates end of packet. The flit which “BoP” flag is ‘1’ is the header and contains the destination address of the packet. Also a flit which “EoP” is ‘1’ is the packet’s trailer. In addition, in an input (input ‘i’) port when a flit of a packet arrives in the FIFO, “ROK” becomes ‘1’ which means the input FIFO is not empty. Therefore, if both the “BoP” and “ROK” signals are ‘1’, routing modules performs the routing to indicate the packet’s output port (output ‘j’) by sending a request

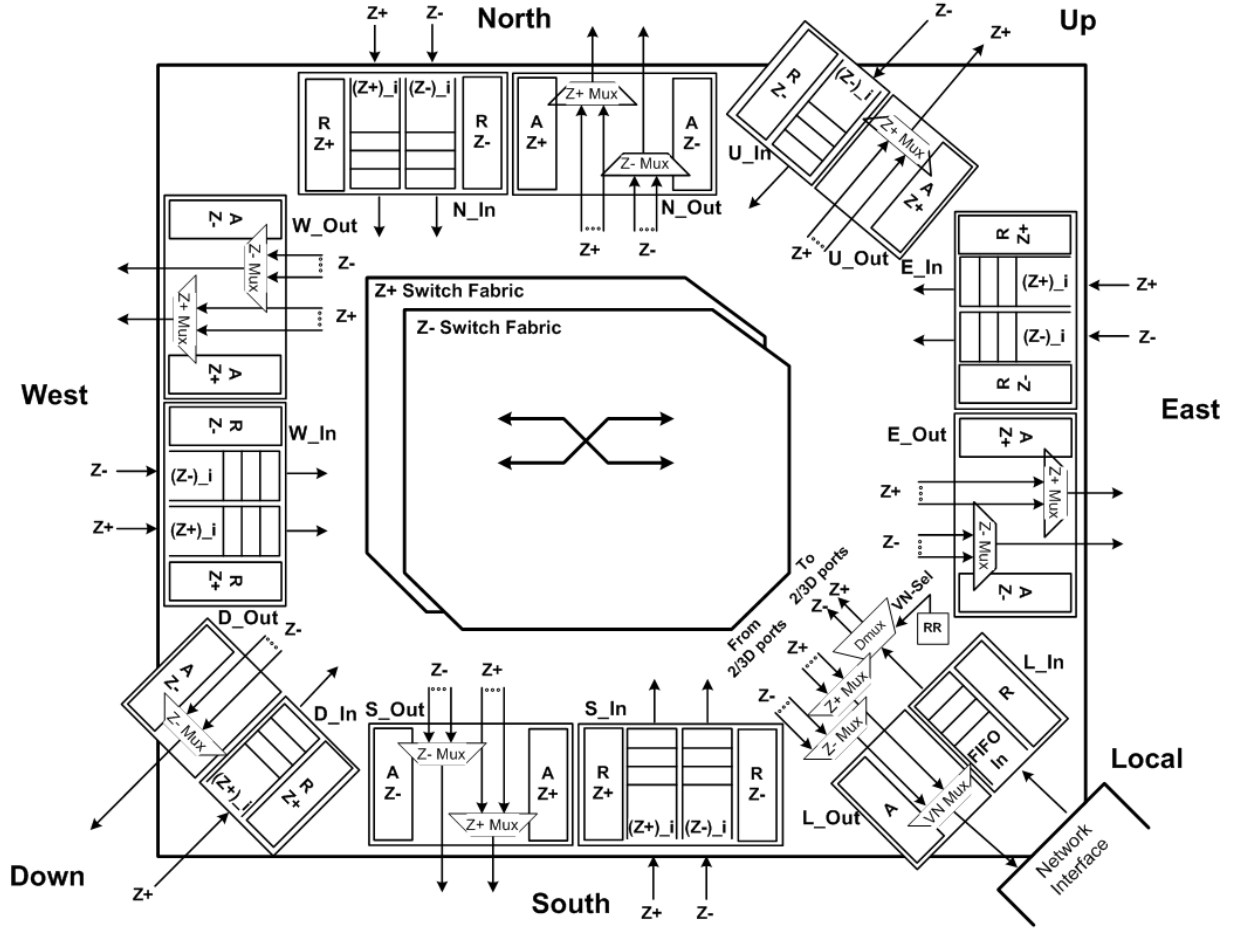


Figure 5.3: Elevator-First Router

“Req_{i,j}” to the output arbitration, as can be seen in Figure 5.2. The arbitration module in the output port selects one request among the received ones based on round-robin order, and enables the ready signal “Rdy_{j,i}” which is the select input of the output ‘j’ multiplexers to show its readiness to receive the packet’s flits. In this way, the output port ‘j’ is allocated to the input port ‘i’ until the end of the packet.

2. Elevator First Router

In Figure 5.3, the block diagram of Elevator First router based on DSPIN router is detailed. The router is composed of three types of ports: 1) 2D ports (NORTH, WEST, SOUTH, EAST), 2) 3D ports (UP and DOWN), and 3) LOCAL port and two Z+ and Z- switch fabrics. The micro-architecture of Elevator First router is based on the characteristics of Vertically-Partially-Connected 3D-NoC and Elevator First routing algorithm. As described before in Chapter Four, to avoid deadlocks, the packets which go to upper layers use the Z+ virtual network, and the packets which go to lower layers use the Z- virtual network. From the router’s point of view, the DOWN input port which receives the packets from lower layer and the UP output port which sends the packets to the upper layer belong to Z+ virtual network. Similarly, the UP input port which receives the packets from the upper layer, and the DOWN output port, which sends the packets to the lower layer belong to Z- virtual network. Each of the 2D ports virtually has two independent Z+ and Z- ports which can

Table 5.1: The connections of Z+ and Z- Switch Fabrics;
(VN+/-: Virtual Network Z+/Z-); (SF+/-:Switch Fabric Z+/Z-); (×: no connection)

In	Out	N		W		S		E		U	D	L
		VN +	VN -	VN +	VN -	VN +	VN -	VN +	VN -			
N	VN +	×	×	×	×	SF+	×	×	×	SF+	×	SF+
	VN -	×	×	×	×	×	SF-	×	×	×	SF-	SF-
W	VN +	SF+	×	×	×	SF+	×	SF+	×	SF+	×	SF+
	VN -	×	SF-	×	×	×	SF-	×	SF-	×	SF-	SF-
S	VN +	SF+	×	×	×	×	×	×	×	SF+	×	SF+
	VN -	×	SF-	×	×	×	×	×	×	×	SF-	SF-
E	VN +	SF+	×	SF+	×	SF+	×	×	×	SF+	×	SF+
	VN -	×	SF-	×	SF-	×	SF-	×	×	×	SF-	SF-
U		×	SF-	×	SF-	×	SF-	×	SF-	×	SF-	SF-
D		SF+	×	SF+	×	SF+	×	SF+	×	SF+	×	SF+
L		SF+	SF-	SF+	SF-	SF+	SF-	SF+	SF-	SF+	SF-	×

transmit both packets belong to whether Z+ or Z- virtual network. To summarize, we call Z+ virtual network, VN+, and Z- virtual network, VN- . As the packets of each virtual network should not use the other virtual network to reach their destinations, we define two different switch fabrics Z+ and Z- (SF+ and SF-) in the Elevator First router to completely separate the Z+ and Z- traffics. The switch fabric Z+ (SF+) connects the following ports to each other:

- The DOWN input port
- Th UP output port
- The Z+ 2D in/output ports
- The LOCAL in/output port

Symetrically, the switch fabric Z- (SF-) connects the following ports to each other:

- The UP input port
- The DOWN output port
- the Z- 2D 2D in/output ports
- the LOCAL 2D in/output ports

Table 5.1 shows the details connections of both Z+ and Z- switch fabrics. As can be seen, not only there is no connection between the Z+ and Z- switch fabrics, but also there is no connection between North/South input ports and East/West output ports, due to the fact that the dimension order deadlock free X-First routing algorithm has been utilized for intra layer packet routing. Indeed, the packets firstly are routed in X direction to be aligned with the Y part of their destination, and then, in Y direction to reach to the destination, as explained in Chapter Four.

3. Addition and Removal of Temporary Header in Elevator First Router

In Vertically-Partially-Connected 3D-NoC, when a 2D node wants to communicate with another node in another layer, the Elevator First routing algorithm forwards the 2D node's packets to its assigned Elevator by adding a temporary header which contains the address of the Elevator. When the packets reach to the Elevator, the temporary header has to be removed and the packet is forwarded to a proper 3D port. In Figure 5.4, an example is presented, when

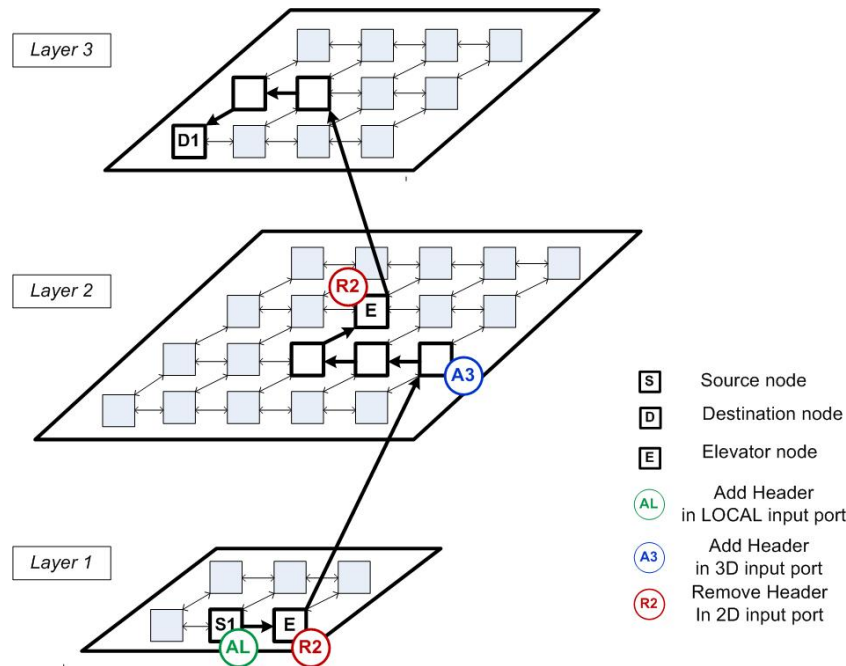


Figure 5.4: Addition and removal of the temporary headers in Elevator First routing algorithm

S1 in layer one wants to send a packet to D1 in layer three. The LOCAL input port checks the destination layer of the packet which is Layer 3 here. Therefore the LOCAL input port needs to inject the packet to the Z+ virtual network and forwards it to the assigned up-Elevator and adds a Z+ temporary header. The packet reaches the up-Elevator always through a 2D input port since it travels in the plane, which is the EAST Z+ input port in the example. In the up-Elevator, the temporary added header is removed in the 2D input port and always forwarded to a 3D output port which is UP output port in Figure 5.4. In layer two, as an intermediate layer, when the packet is received by the DOWN input port, a new temporary header is added to the packet since the current layer is not the destination layer and the current node is a 2D node. Then, the packet is forwarded to a 2D output port, which is EAST Z+ output port in the example, to reach to its new up-Elevator. Finally, the 2D input port (SOUTH Z+) in the up-Elevator in Layer 2 receives the packet, removes the temporary header, and then forwards it to the UP output port to reach its destination layer.

Consequently from a micro-architectural point of view, adding a temporary header is always done in LOCAL and 3D input ports and removing a temporary header is always done in 2D input port. Since each 2D input port has two separated Z+ and Z- virtual networks, 2D Z+ input port removes the temporary headers of Z+ virtual network's packets and 2D Z- input port removes the temporary headers of the Z- virtual network's packets. Moreover, as the UP input port belongs to Z- virtual network, it adds the Z- temporary header to Z- virtual network's packets and as the DOWN input port belongs to Z+ virtual network, it adds the Z+ temporary header to Z+ virtual network's packets. Unlike 2D and 3D input ports, LOCAL port belongs to neither Z+ nor Z- virtual network. As LOCAL port is completely independent, addition of both Z+ and Z- temporary header is possible in LOCAL input port.

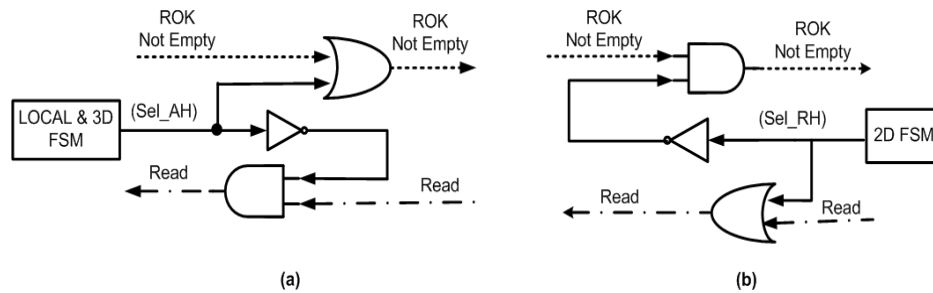


Figure 5.5: a) Addition the temporary header b) Removal the temporary header

It is determined based on the position packet's destination layer compared to the position of packet's source layer.

Since the ROK signal (write on downstream router's FIFO) means the input FIFO is not empty and the WOK signal (read from current router's FIFO) means the input FIFO of downstream router is not full, with a very simple manipulation of these signals (by only one add or remove command signal generated by a FSM in each port and three simple gates) the procedures of addition and removal of a temporary header can be realized. In Figure 5.5.a the addition circuit and in Figure 5.5.b the removal circuit are shown. In Figure 5.5.a when the Elevator First FSM of the 3D and LOCAL ports are in Add Header (AH) state, a select signal (Sel_AH) becomes enabled. This signal does not allow the Read signal to reach the input FIFO, although ROK signal is enabled. Therefore, the input port can write the temporary header as the first flit of the packet to the input FIFO of downstream router. In this case one extra cycle is consumed to add the temporary header. In Figure 5.5.b when the Elevator First FSM of the 2D ports is in Remove Header (RH) state, a select signal (Sel_RH) becomes enabled. This signal does not allow the ROK signal to reach to the input FIFO of downstream router, although Read signal is enable. Therefore, the input port does not write the temporary header as the first flit of the packet to the input FIFO of downstream router. In this case also one cycle is consumed to remove the temporary header.

4. The Micro-Architecture of the Ports in Elevator First Router

In this section, the micro-architecture of the ports in Elevator First router is explained in detail. Since the LOCAL port is the entrance gateway of the packets to Vertically-Partially-Connected 3D-NoC, it is explained first. LOCAL port is the most complicated port in Elevator-First router architecture. This is because of two reasons. The first reason is that the LOCAL input port is the only port that should decide upon the packet's virtual network. The second reason is the arbitration logic in the LOCAL output port. The arbitration module not only should decide the next input port to be served based on the previous status, but also must indicate which of the virtual networks (Z+ or Z-) should be served.

When a packet is received by LOCAL input port, the type of its virtual network is not clear. Therefore, LOCAL input port defines the packet's virtual network type. If the source and destination layers are different, the Z+ network will be selected if the packet has to go up, and the Z- network will be selected if the packet has to go down. Otherwise (the source and destination are on the same layer) the packet has to be forwarded to the virtual network which had not been selected for the previous packet. The LOCAL finite state machine is depicted in

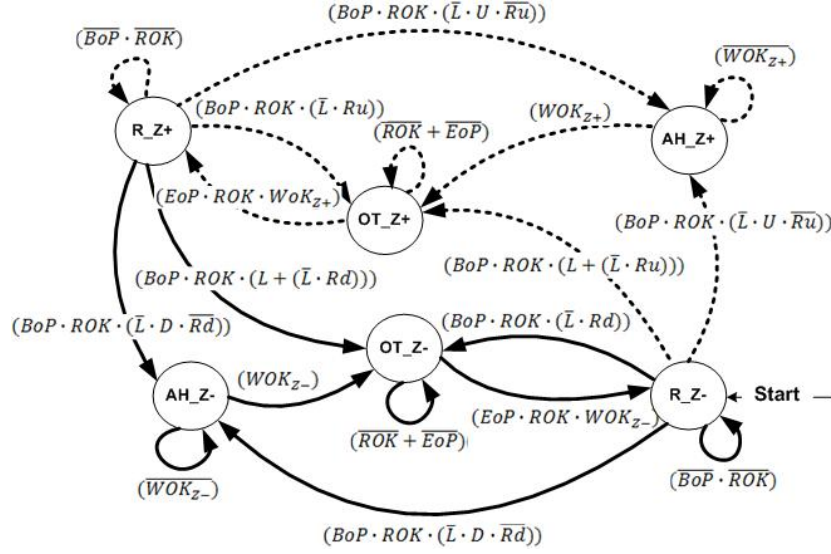


Figure 5.6: LOCAL Elevator First Finite State Machine

Figure 5.6. As can be seen, the start state is R_Z- (return from Z-) which based on round robin order means if the source and destination of the packet are in the same layer ($L = 1$), the packet will be forwarded to the OT-Z+ (Ordinary Transmission in Z+ network). On the other hand, if the packet's destination is in the upper (U) or lower layer(D), the Z+ or Z- virtual network will be selected. Therefore, the R_Z- state will be changed to OT_Z+ or OT_Z- respectively if the current node is its up or down-Elevator. If the request signal to UP output port (Ru) or DOWN output port (Rd) is enabled, it means that the current node is its own Elevator. Otherwise the R_Z- will be changed to AH_Z+ (Add Header in Z+ network) or AH_Z- (Add Header in Z- network).

Therefore the LOCAL input port can send the packet to either Z+ or Z- virtual network based on the packet's destination address. In order to do this, the LOCAL port contains only an Elevator First routing algorithm module, a FIFO, and an Elevator First finite state machine to decide upon the type of virtual channel the packet should go into and add a temporary header. Output port is symmetrically similar to input port. Therefore, both Z+ and Z- data and control signals are connected to LOCAL output port which makes the LOCAL arbitration module and multiplexers more complex than the output ports of other ports. Figure 5.7 depicts the micro-architecture of the LOCAL input port. LOCAL Elevator First FSM in LOCAL input port not only enables the command signal of Sel_AH to manipulate the Write and Read signals when adding a new header, but also chooses the proper virtual network for a new arrival packet with the VN_Sel signal (See Figure 5.7). When Sel_AH is enabled, the input port buffer cannot receive the read request signal, so the permanent header of the packet will not be removed. Concurrently, Sel_AH enables the write request signal of the next router, so the temporary header will be written instead of the permanent header. Sel_AH should be disabled when the next input port buffer successfully receives data, i.e. just in the next cycle after when WOK (Read) is enable. In the next clock cycles the permanent header and the rest of the flits until the end of the packet will be transferred to the 2D output port.

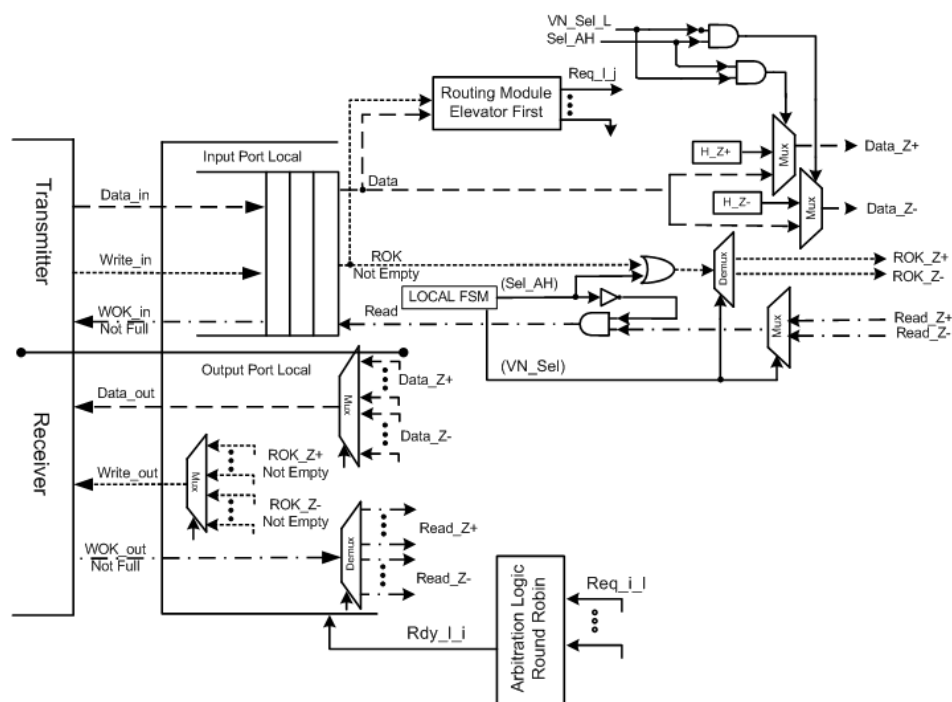


Figure 5.7: Micro-architecture of LOCAL Elevator First input port

Fundamentally the LOCAL Elevator First finite state machine is composed of both UP and DOWN Elevator First finite state machines, as can be seen in Figure 5.8.b and 6.9.b. Like LOCAL input port, the addition procedure is the same in 3D input ports. A point which has to be mentioned is that the input UP port adds down-Elevator temporary header to the packets of Z- virtual network and the input DOWN port adds up-Elevator temporary header to the packets of Z+ virtual network (see Figure 5.8.a and 5.9.a respectively).

As the two virtual networks are completely separated from each other to avoid deadlock, a 2D port is separated into two virtual networks of Z+ and Z-. In Figure 5.10 the micro-architecture of 2D ports is detailed. Each of Z+ and Z- input ports contains an Elevator First routing module, a FIFO, and an Elevator First finite state machine to remove the temporary header of the packets. Also each of Z+ and Z- output ports contains a round robin arbitration module and the multiplexers to multiplex and demultiplex the data and control signals of the connected input ports via switch fabric.

In Figure 5.11, the 2D Elevator First finite state machine is presented. When T flag in the packet's header is '1', it means that the header is temporary. In addition, if one of the 3D output port (Ru (request to UP output port) and Rd (request to DOWN output port)) is selected it means that the current node is Elevator and the packet should be removed. Removing the temporary header can be simply done by enabling Read to read the first flit of the packet and disabling Write to not write in the 3D output port (in fact in the buffer of the 3D input port of the next router). Sel_RH_Z+/Z- are the command signals generated by 2D_FSM_Z+/Z- finite state machines to remove the header. They will be enabled just for one clock cycle. By disabling them in the next clock cycle the real (permanent) header (the second flit of the packet) will be transferred by the 3D output port and the flit transferring process will proceed until the end of the packet.

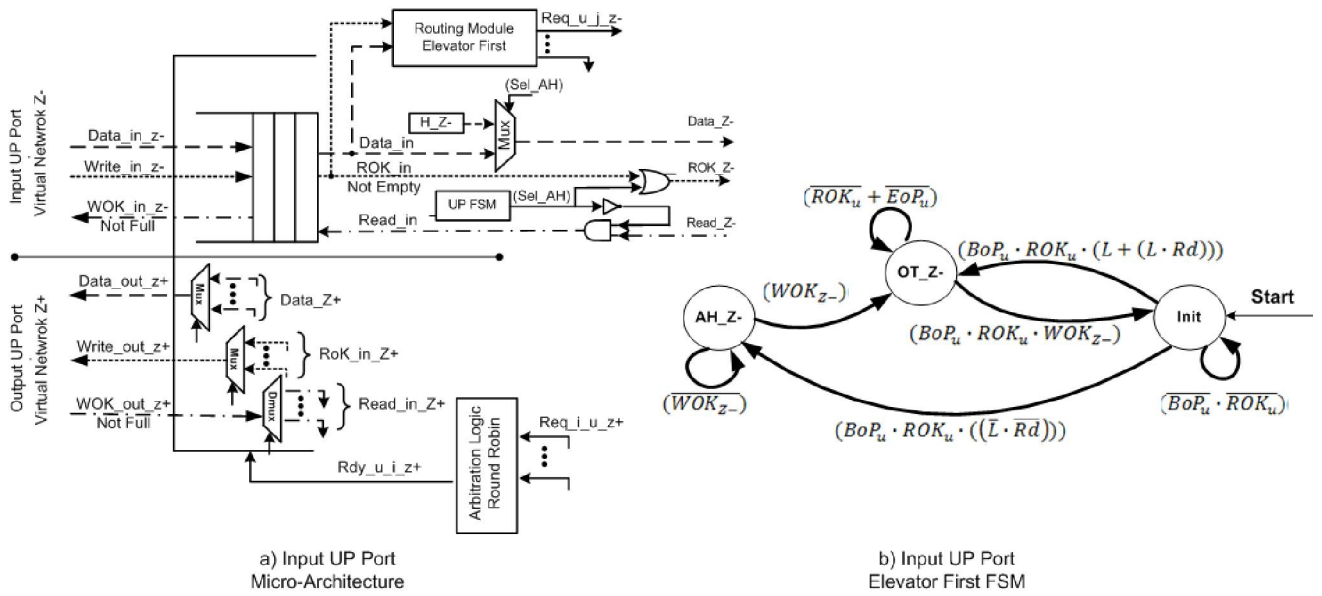


Figure 5.8: UP_Elevator First input port

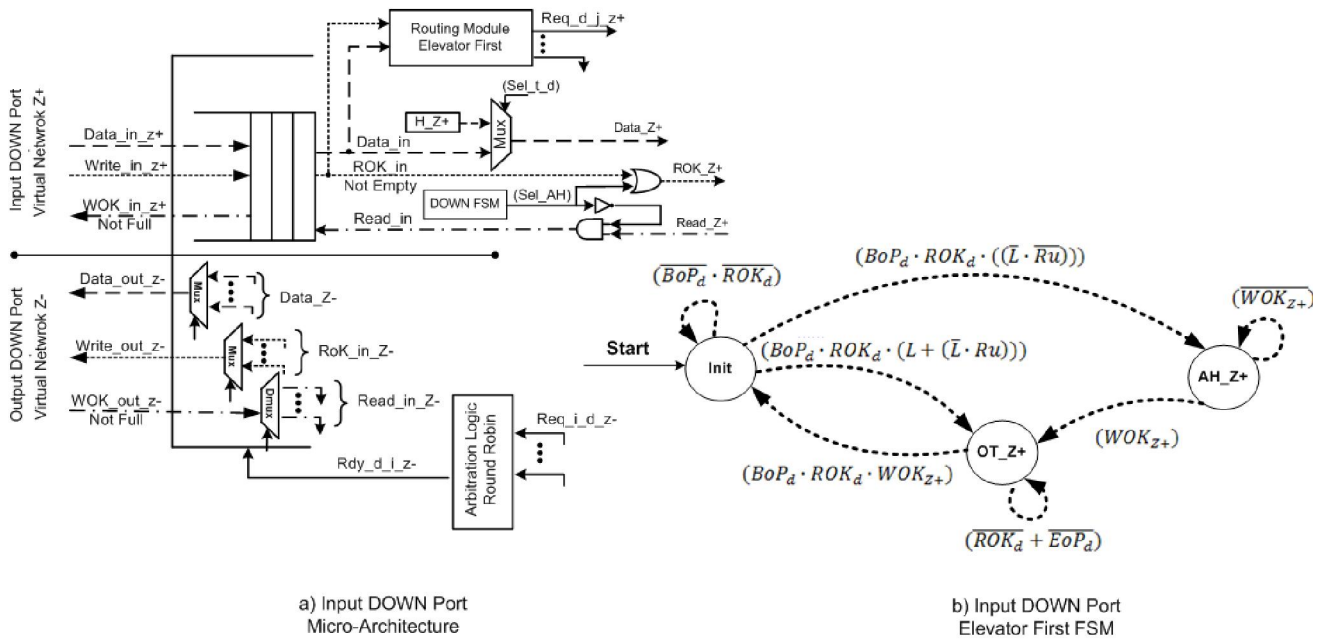


Figure 5.9: DOWN_Elevator First input port

CHAPTER FIVE: MICRO-ARCHITECTURE OF ELEVATOR FIRST ROUTER

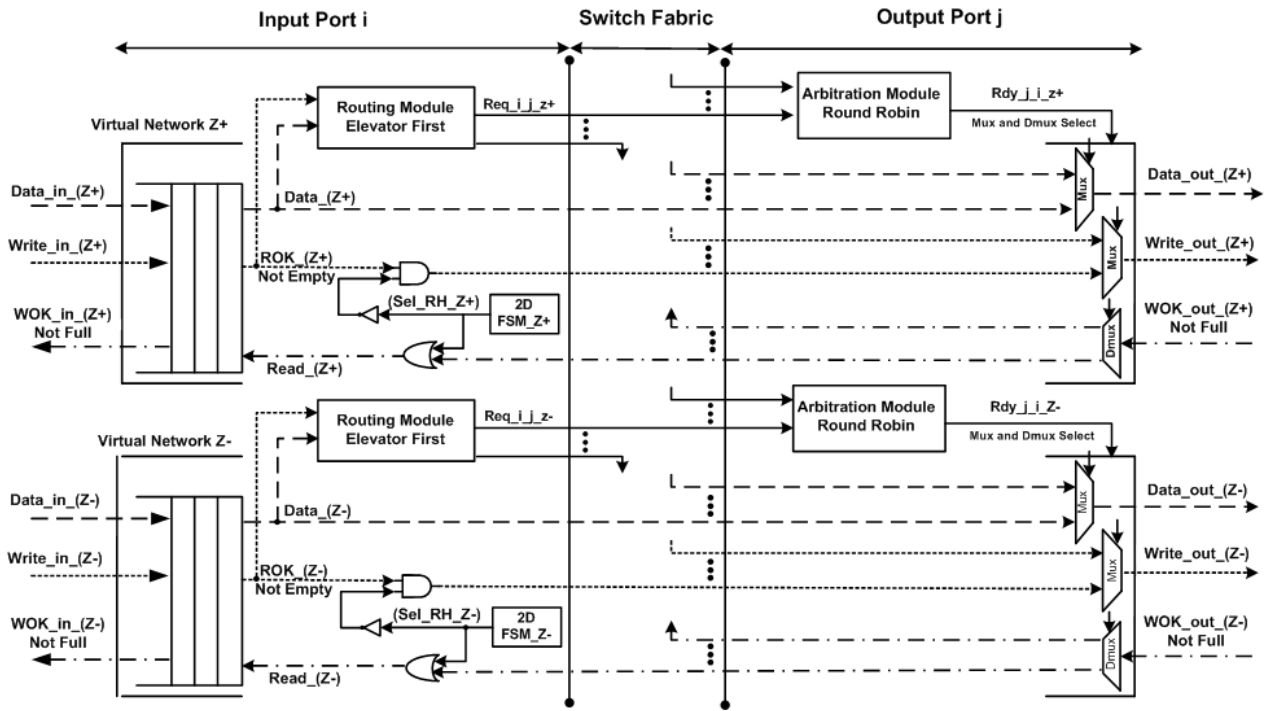


Figure 5.10: 2D Elevator First input port

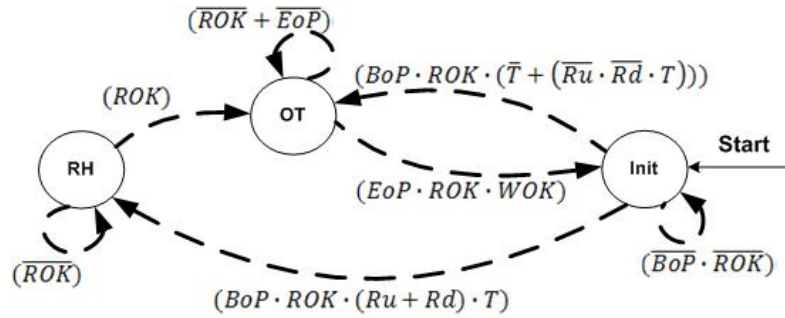


Figure 5.11: 2D Elevator First Finite State Machine

5. RTL-Level Synthesis

We have developed Elevator-First 3D router in VHDL and synthesized it with Synopsys using CMOS 65nm technology of STMicroelectronics (GP65LVT). We synthesize the router with 0ns constraint to achieve the maximum operational frequency. In Table 2 we show the area, the maximum clock frequency, and the power consumption of an Elevator-First 3D router with FIFO depth of 8 flits of 34 bits (32 data bits and 2 control bits). In this table 5-Port router represents the 2D router without any vertical port, 6-Port represents a 3D router with Up or Down vertical port, and 7-Port represents a 3D router with both of Up and Down vertical ports.

As final point, in order to comprehend the area overhead of Elevator-First router Figure 5.12 depicts its silicon area as well as the area of an ordinary 3D router with 7 ports, and that exploits larger FIFOs (in which the buffering capacity is approximately equivalent to

CHAPTER FIVE: MICRO-ARCHITECTURE OF ELEVATOR FIRST ROUTER

Table 5.2: Area of different types of Elevator-First Router

Type	Total area	FIFOs area	Max Freq.	Power Cons.
5-Port	57754 μm^2	45713 μm^2	3.7 GHz	2.5 mW
6-Port	65053 μm^2	49762 μm^2	3.4 GHz	3 mW
7-Port	72087 μm^2	53554 μm^2	3.2 GHz	3 mW

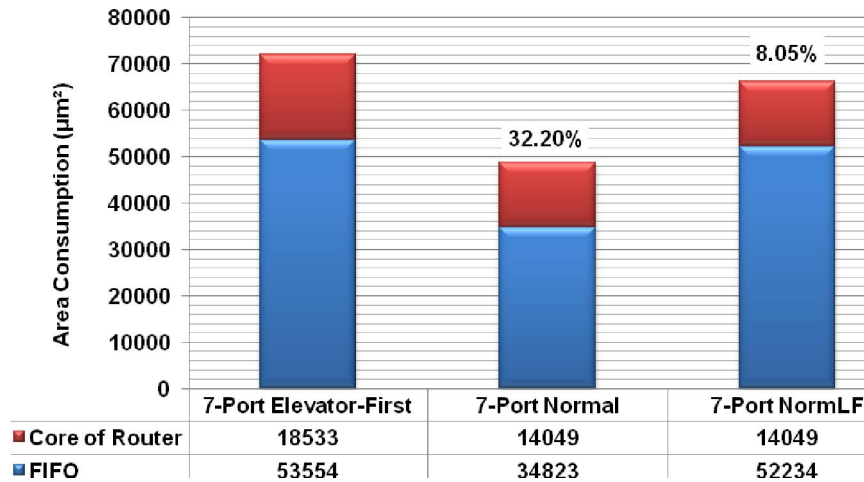


Figure 5.12: Router Area Comparison

that of Elevator-First router, i.e. UniNormLF or LocNormLF). As can be seen the Elevator-first router area is about 32% larger than an ordinary fully connected router, but the area overhead, compared to an equivalent router in terms of buffer capacity, is only 8%.

6. Conclusion

This chapter details the design and implementation of Elevator First router based on Elevator-First distributed deadlock-free routing algorithm, which has been presented in Chapter Four, to be used in Vertically-Partially-Connected 3D-NoCs. The Elevator-First router contains two virtual networks (Z^+ and Z^-) to cope with deadlocks. We have also evaluated the practicality of the design in terms of performance (average latency, average hop-count, and operating frequency) and cost (area and power consumption). We have synthesized the RTL model of the router architecture using GP65LVT technology of STMicroelectronics. Its area overhead compared to a standard fully connected 3D-NoC router with an equivalent amount of buffer capacity is only about 8%.

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NoC

In the “Problem Definition Chapter”, we explained the performance degradation problem in Vertically-Partially-Connected 3D-NoC due to the removed inter layers links (vertical links). Also we described that this problem can be taken into account in both micro-architecture (router hardware implementation) and system (topology design) levels. We have already demonstrated the micro-architecture of Elevator First router in Chapter Five. The architectural exploration of Vertically-Partially-Connected 3D-NoC is discussed in this chapter.

As we showed in “Problem Definition Chapter”, the number of possible topologies in Vertically-Partially-Connected 3D-NoC is huge. In addition, many parameters affect the performance of Vertically-Partially-Connected 3D-NoC such as average hop-count, the number of nodes the Elevators serve, and etc. These issues make the analysis of the architectures difficult. In this chapter we

- I. define and study the parameters which describe the performance of Vertically-Partially-Connected 3D-NoC,
- II. discuss and evaluate a placement and assignment method which
 - i. guarantees minimum possible hop-count between the 2D nodes and their Elevators,
 - ii. uniformly distributes the Elevators in the layers, and
 - iii. tries to equally assign the 2D nodes to the Elevators, and finally
- III. present an estimation quadratic-based method to estimate the saturation threshold of Vertically-Partially-Connected 3D-NoC.

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOc

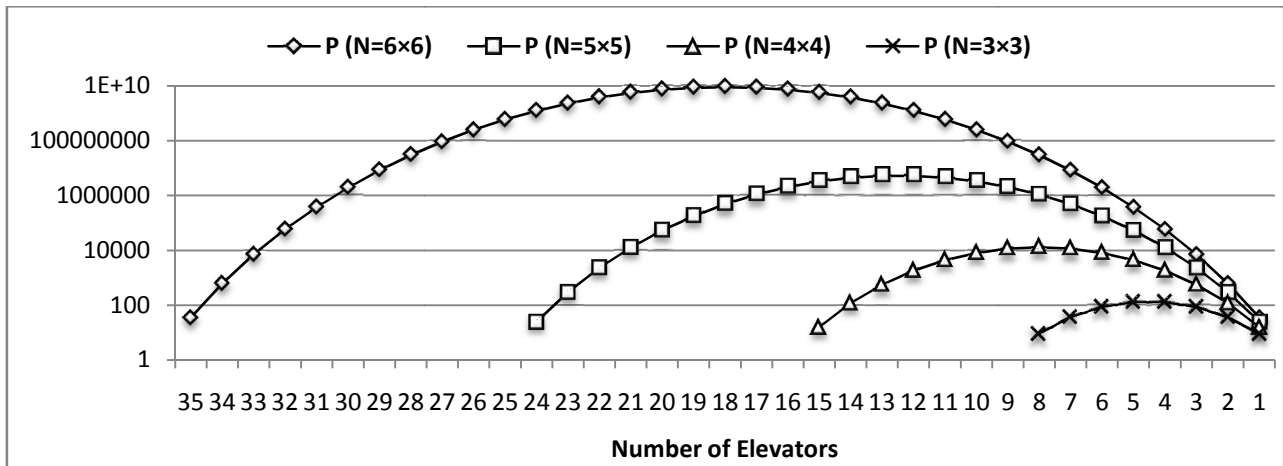


Figure 6.1: Number of vertical links placements in a Mesh-based layer; $P = \binom{N}{E} = \frac{N!}{E! \times (N-E)!}$

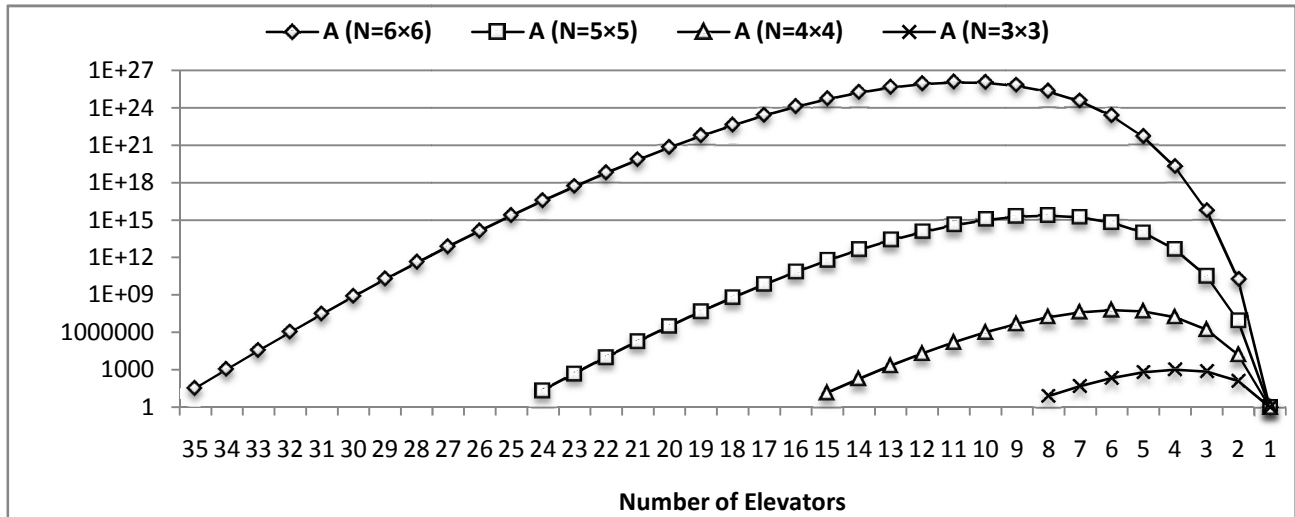


Figure 6.2: Number of 2D nodes assignments in a Mesh-based layer; $A = E^{(N-E)}$

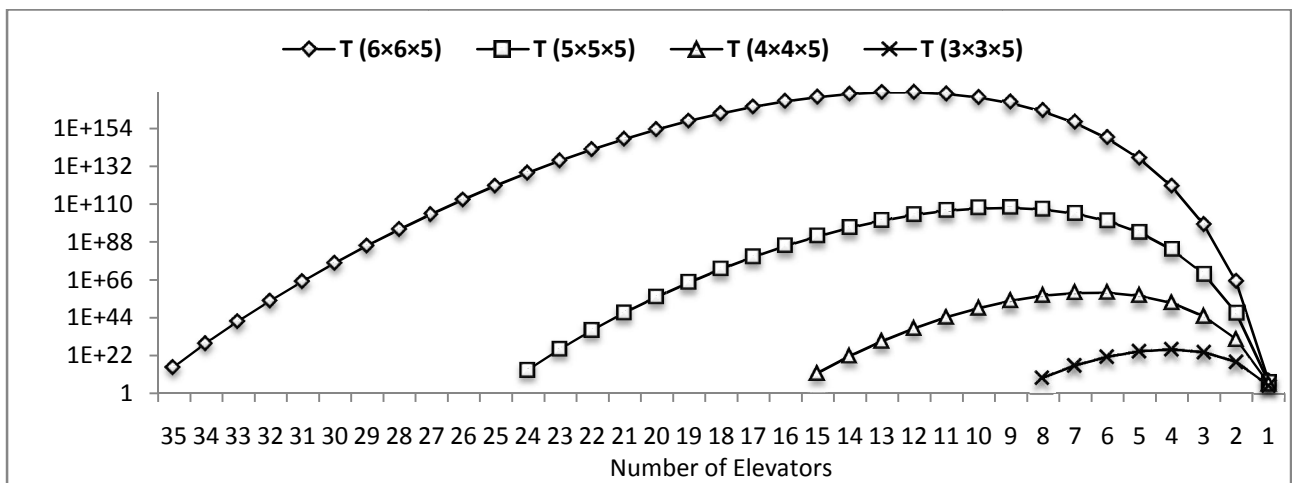


Figure 6.3: Number of topologies in Vertically-Partially-Connected Mesh-based 3D-NOc; $T = (P \times A)^Z$

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NoC

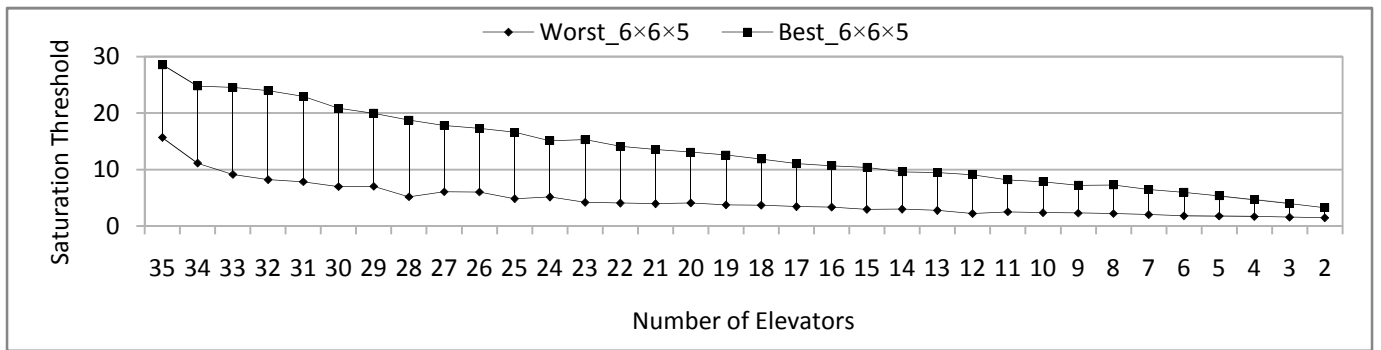


Figure 6.4: The gap between the best and worst saturation thresholds of 18.5 K topologies for different number of Elevators in 6x6x5 network

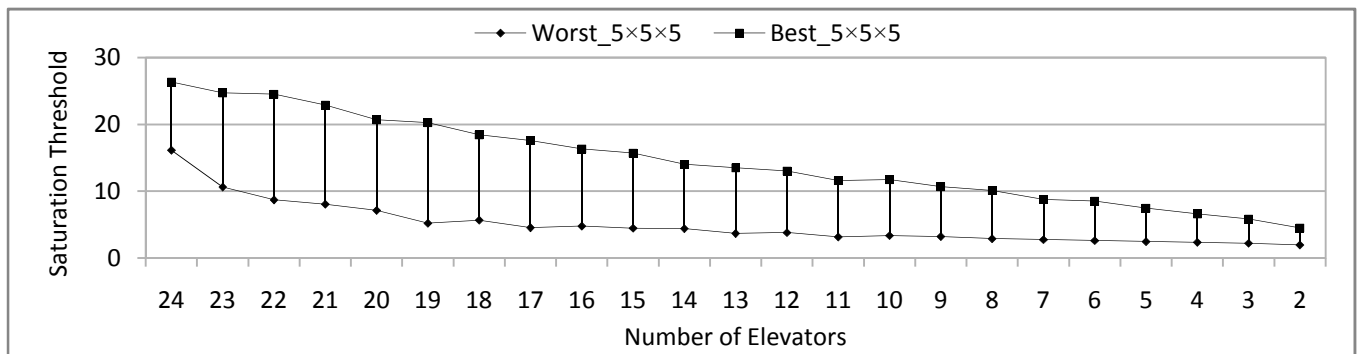


Figure 6.5: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in 5x5x5 network

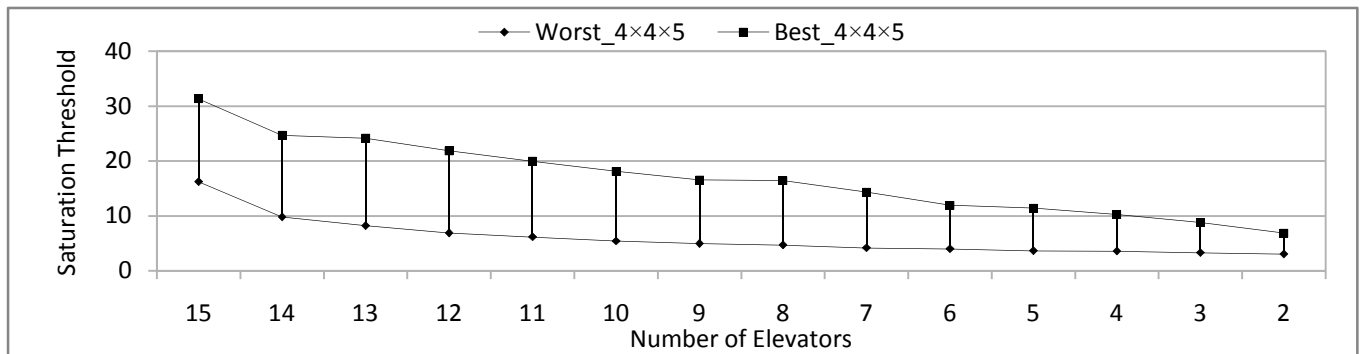


Figure 6.6: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in 4x4x5 network

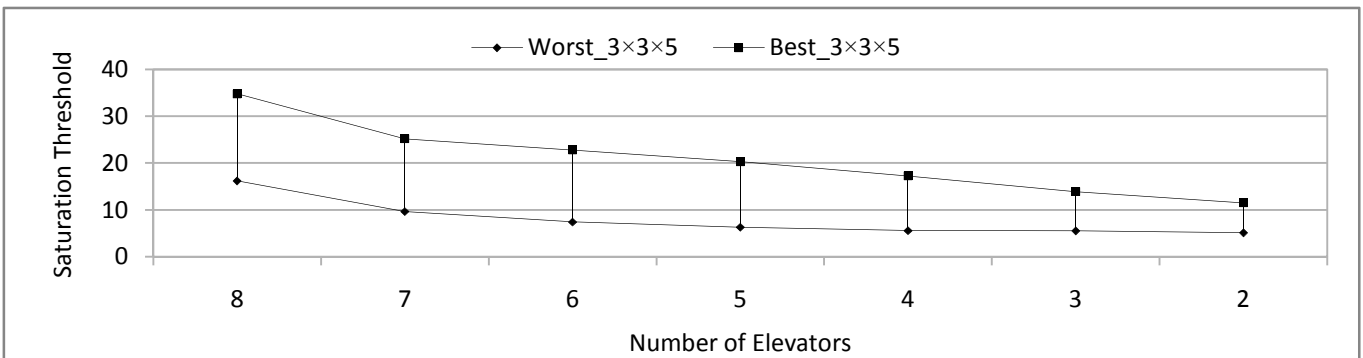


Figure 6.7: The gap between the best and worst saturation threshold of 18.5 K topologies for different number of Elevators in 3x3x5 network

1. Analysis of Number of Possible Topologies in Vertically-Partially-Connected 3D-NoC

The possible number of topologies in one layer equals the multiplication of the possible number of placements (i.e. position of Elevators in layer) and assignments (i.e. which Elevator is used by a given node to send its packets to upper and lower layers). In Figure 6.1 and Figure 6.2, the possible placements and assignments of Elevators (E) (vertical links) in one layer with different size (number of nodes, N) are plotted. As can be seen, the number of assignments is larger than the number of placements for each (N, E) set. Also, the number of possible topologies and the number of (N, E) sets increases exponentially with N . The possible topologies for different layer sizes (9, 16, 25, 36) and 5 layers are also depicted in Figure 6.3. As can be seen, the number Z used as exponent, which is the number of layers, raises the number of possible topologies significantly. Of course in the equation of number of possible topologies, we assume that every layer can have different placements and assignments for the same number of (N, E) .

We generated 18500 samples of topologies (with random Elevator placements and assignments) and calculated their saturation thresholds of average latency under uniform traffic pattern with an analytical tool [71] for different number of Elevators. The results of the best and worst saturation thresholds for each (N, E) set of 18500 topologies are shown in Figure 6.4, 6.5, 6.6, and 6.7 for network sizes equal to $(6 \times 6 \times 5)$, $(5 \times 5 \times 5)$, $(4 \times 4 \times 5)$, and $(3 \times 3 \times 5)$ respectively. The gap between the best and worst saturation threshold in all the network sizes is considerable. Therefore, good placement and assignment of Elevators in the layers can lead to a topology which can tolerate more traffic load with a smaller number of Elevators. In the next section, we define some parameters which have effect on the performance of Vertically-Partially-Connected 3D-NoC.

2. Region Concept

Each node in Vertically-Partially-Connected 3D-NoC is assigned to an UP-Elevator (Elv_u) and a DOWN-Elevator (Elv_d) which are used to forward its packets to upper layers and lower layers respectively. We define a “Region” as a set of an “Elevator” (UP or DOWN) and its “assigned 2D nodes”. Therefore, in each layer, the numbers of Elevators (E) and Regions (R) are equal. In Figure 6.8, the concept of Region in Vertically-Partially-Connected 3D-NoC architecture is presented, the address of each node being shown in (XYZ) order. Since there are two main types of Elevators (UP and DOWN), there are two types of Regions as well: up-Region (ur) and down-Region (dr) (see Figure 6.8). Based on the Regions concept, four main parameters can be defined to show the performance and structure of a Region(r_i):

Region Degree(RD): Region-Degree is the number of nodes in a Region which are served by the Region Elevator. Note that the Elevator itself is a router which is connected to an IP core, therefore it is also part of the Region. For example, as can be seen in Figure 6.8, the up-Region ‘1’, in Layer ‘1’, contains (211, 111, and 121) nodes when the node (111) is the up-Elevator of the up-region ‘1’. Therefore the Region-Degree of the Region ‘1’ is 3.

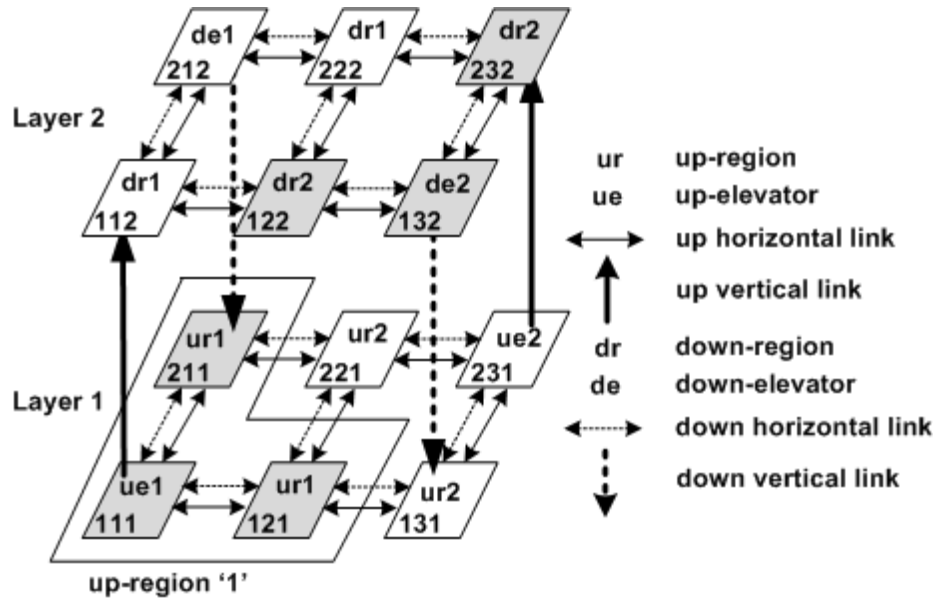


Figure 6.8: Region concept in Vertically-Partially-Connected Mesh-based 3D-NoC

Region Average Hop-Count(RAHC): Hop-count affects the performance (average latency) and power consumption of NoC. In the Region(r_i), Region Average Hop-Count is the average of Manhattan distances between each member (m_{ij}) and the Region Elevator (E_i). As can be seen in Figure 6.8, in up-region '1', the Region Average Hop-Count is $\frac{1+0+1}{3}$ which equals 0.66.

Region Load(RL): The load of a link is related to both the routing algorithm and traffic pattern. We calculate the loads of all the links based on uniform distribution, under Elevator First routing algorithm. For each region(r_i), the Region-Load is the summation of the loads of all the links of Manhattan distances between each member of the Region and the Region Elevator. In Figure 6.9, the Region-Load of the Region 'i' in layer 3 equals ($dLi_1 + dLi_2$), where dLi_1 and dLi_2 are the load of the links between the Region Elevator (de_i) and the nodes $dr1$ and $dr2$ respectively.

Total Region Degree(TRD): This parameter is the total number of nodes an Elevator has to serve. Two connected Regions in adjacent layers are vertically connected via Regions' Elevators. For example, as can be seen in Figure 6.9, a packet from the third layer has to pass via the second layer to reach to the first layer. Figure 6.9 demonstrates the concept of Total Region Degree. As can be seen, the Region-degree of Region 'j' in second layer is 2. While the Total Region-Degree of Region 'j' is 5 which is the summation of the Region-Degrees of both Region 'i' in third layer and 'j' in second layer. In other words, the Elevator of Region 'j' in second layer not only has to serve the packets of its own Region in second layer but also the Region 'i' in third layer either. In the following, these parameters are used to study and estimate the performance of Vertically-Partially-Connected 3D-NoC.

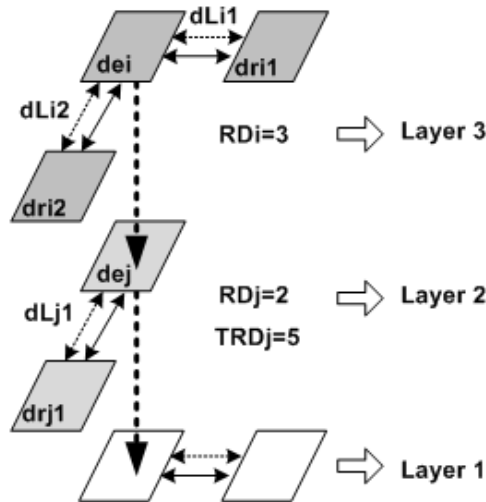


Figure 6.9: The Load & Total Region-Degree

3. Pattern-based Vertically-Partially-Connected Mesh-based 3D-NoC Placement and Assignment Algorithm

Hop-count is one of the important issues in NoC design. Less hop-count means less latency and less power consumption which finally ends in better performance. Elevator First routing in Vertically-Partially-Connected 3D-NoC causes more hop-count, which is of course inevitable, due to the removed vertical links. However, if in each layer the Elevators are placed so that they can be assigned to the 2D nodes with only one hop-count, the average hop-count will be as small as possible. Fundamentally there are $(4 \times \sum_{i=1}^{HP} i)$ nodes which can be assigned to an Elevator with maximum hop-count (HP). Figure 6.10.a shows the one hop-count pattern. In which, there are four possible 2D nodes which can be assigned to an Elevator with maximum one hop-count. Of course, the numbers of 2D nodes which are assigned to an Elevator are decided by the designer and can be less or more than four. If we assume that all the four possible 2D nodes are assigned to an Elevator (Region-Degree equals five) with only one hop-count, then the possible placement of the adjacent Regions can be derived from the reference node which are shown in Figure 6.10.b. If (x_0, y_0) is the address of the primary Region's Elevator:

5. Adjacent North Elevator's address: $(x_0 + (HP), y_0 + (HP + 1)) \xRightarrow{HP=1} (x_0 + 1, y_0 + 2)$
6. Adjacent West Elevator's address: $(x_0 - (HP + 1), y_0 + (HP)) \xRightarrow{HP=1} (x_0 - 2, y_0 + 1)$
7. Adjacent South Elevator's address: $(x_0 - (HP), y_0 - (HP + 1)) \xRightarrow{HP=1} (x_0 - 1, y_0 - 2)$
8. Adjacent East Elevator's address: $(x_0 + (HP + 1), y_0 - (HP)) \xRightarrow{HP=1} (x_0 + 2, y_0 - 1)$

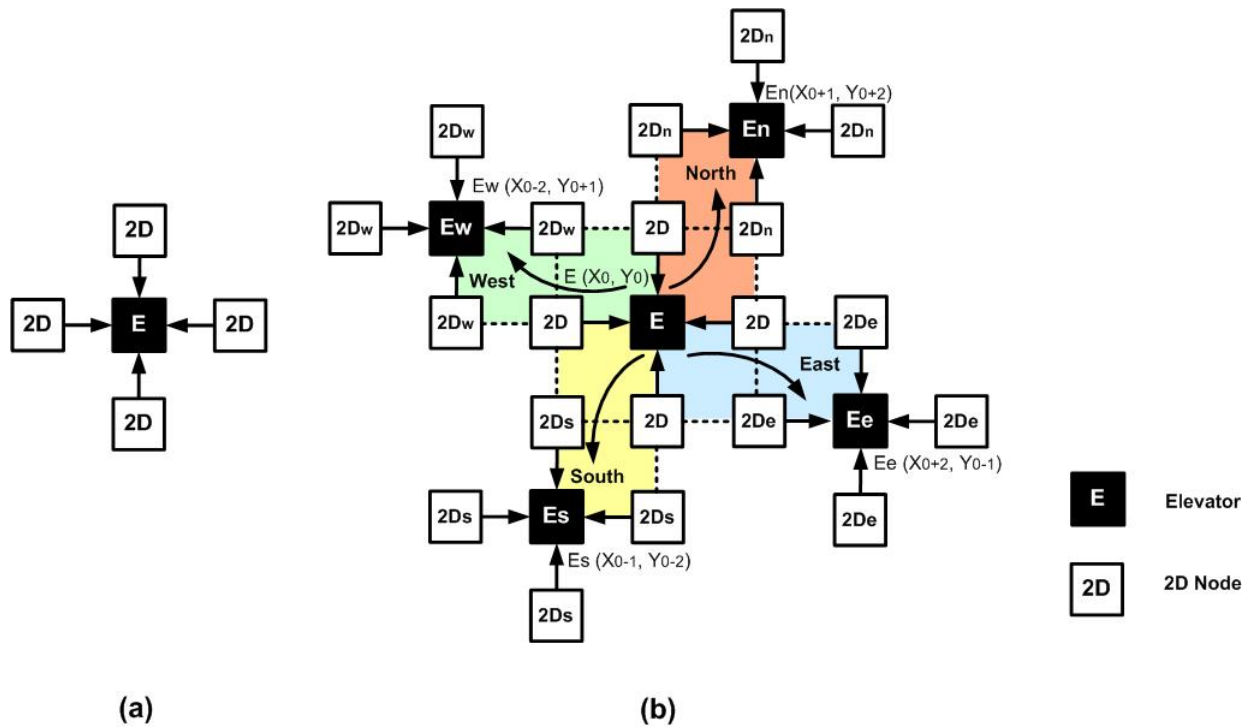


Figure 6.10: Pattern-based Placement and Assignment in Vertically-Partially-Connected 3D-NoC

This approach can be used to place and assign the Elevators to the nodes and make a Vertically-Partially-Connected 3D-NoC topology with similar patterns. Of course the Pattern-based placement and assignment method algorithm that we will explain here is an example. The designer can customize this approach based on the requirements of application and traffic patterns. Therefore, at this stage we assume that all the regions have similar hop-count ($HP = 1$) and Region-Degree ($RD = 5$) for simplicity in explanation and evaluation. Figure 6.11.a depicts the expandability of pattern-based placement in Vertically-Partially-Connected 3D-NoC. The placement and assignment algorithm of the similar Regions with the pattern of four assigned 2D nodes to an Elevator with one-hop can be generalized by the definition of two matrixes (bricks) which are called:

- “WestEast” brick $(x_0 \pm (HP + 1), y_0 \mp (HP))$:
In (west or east) direction and their size is $[(HP + 2) \times (HP + 1)]$
- “NorthSouth” brick $(x_0 \pm (HP), y_0 \pm (HP + 1))$:
In (North or South) direction and their size is $[(HP + 1) \times (HP + 2)]$

Figure 6.11.a shows how these two types of bricks can be placed in a Mesh layer. Also, Figure 6.11.b shows how the placement can be mapped on a Mesh layer. As can be seen in Figure 6.11.b, if we start from a reference node and set its address as the reference Region’s Elevator address (x_0, y_0) , all the Elevators in the layer can be placed based on the referential node. It depends on which side of the Mesh we start, for example if we start from the West side of the Mesh layer, then the Elevators will be placed in East side of the reference node. Therefore we have to know how many bricks we have to pass in East and then in North or South directions. However, the described placement sometimes can not be mapped on a

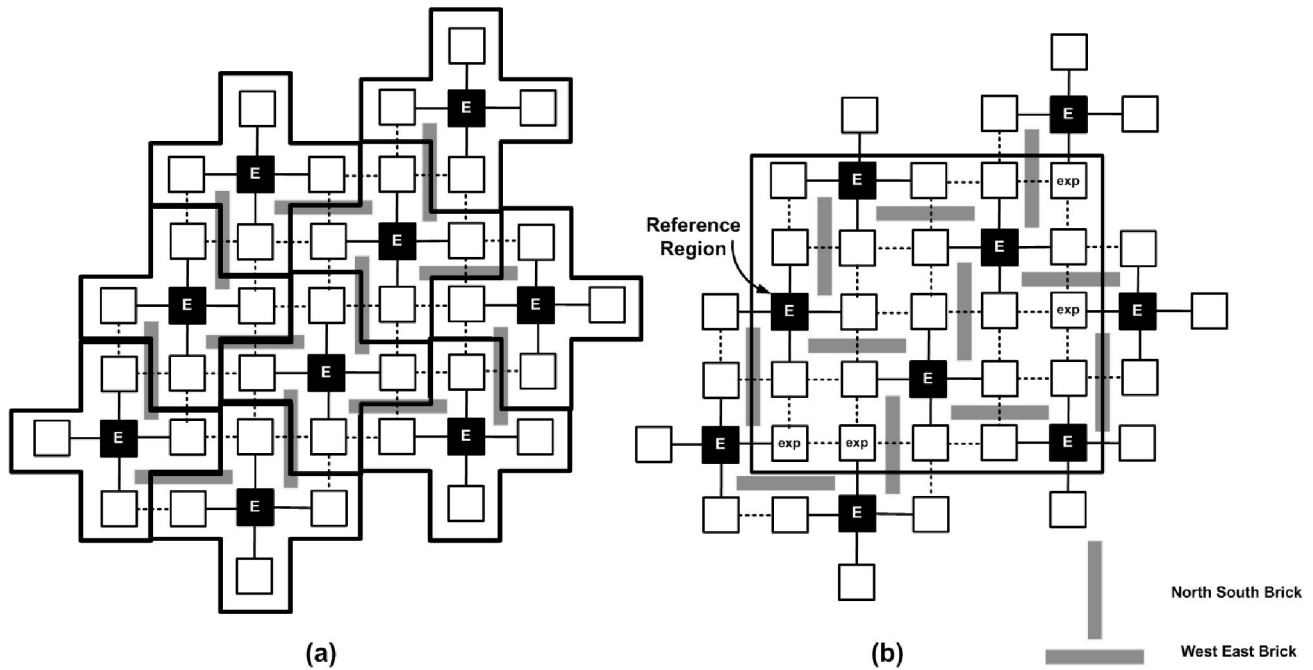


Figure 6.11: Expandability of Pattern-based Placement in Vertically-Partially-Connected 3D-NOC

Mesh-based layer, as can be seen in Figure 6.11.b. The nodes shown “exp” are the 2D nodes which their one hop Elevators are out of the Mesh’s boundary.

Therefore, the pattern-based placement and assignment algorithm has to have two main phases, 1) Initialization phase: to place the Elevators and assign them to the nodes in a Mesh layer based on the reference Elevator’s address in East or West and North or South directions and 2) post-check phase: to check if there are some 2D nodes which are assigned to the Elevators which are not in the Mesh layer territory.

The two phases of the algorithm has been designed to configure each node (x_c, y_c) by the address of the Region’s Elevator of the node. This algorithm just needs the Elevator’s address of the reference Region (x_0, y_0) and the size of the layer $(M \times N)$. Figure 6.12 shows how the initialization phase of pattern based algorithm works. If we assume that the Elevator’s address of the reference Region is $(0,2)$, we just need to know how many bricks are required in (East direction) and (North or South direction). This can be extracted by the calculation of both differences between the X address of the reference and current addresses $(x_c - x_0)$, called Δx_{c0} , and the difference between the Y address of the reference and current addresses $(y_c - y_0)$, called Δy_{c0} , which can show the direction of the line between the reference node and the current node. Based on this, the slope of the line that starts from the reference Elevator and continues in East direction can be calculated as:

$$\frac{\Delta y_{c0}}{\Delta x_{c0}} = -\frac{HP}{HP+1} \quad \text{Equ 1}$$

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOC

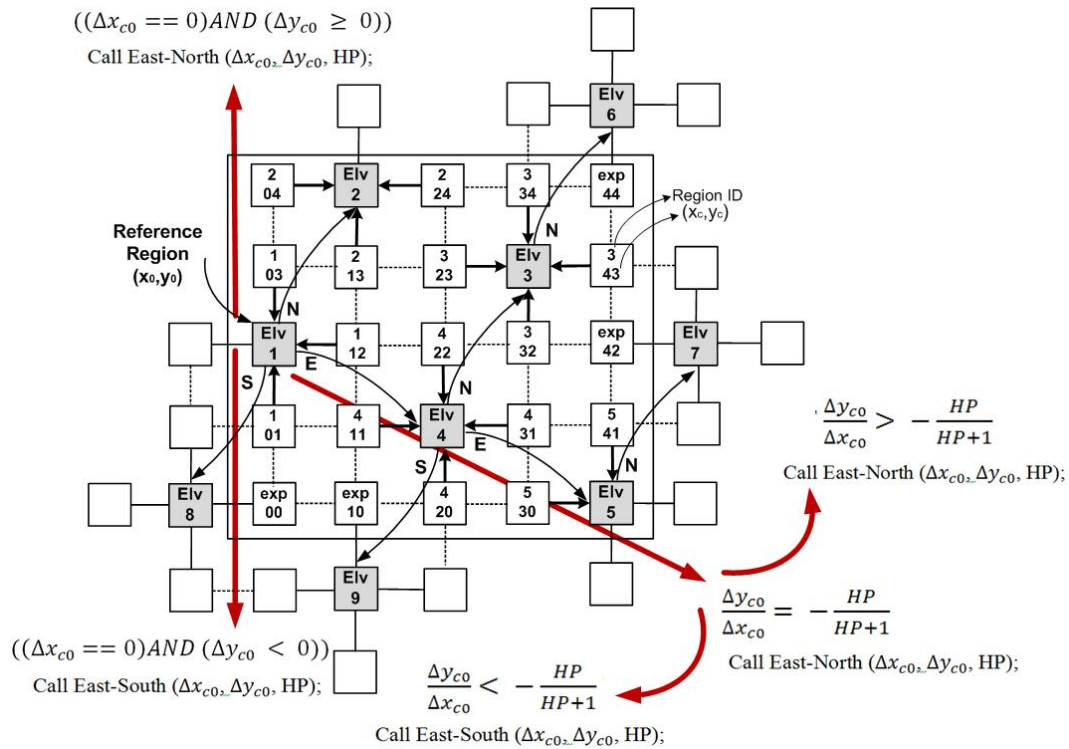


Figure 6.12: The initialization phase of the pattern-based algorithm

Therefore, as also can be seen in Figure 6.12, if $\frac{\Delta y_{c0}}{\Delta x_{c0}}$ of each node (x_c, y_c) is greater than or equals to $(-\frac{HP}{HP+1})$, then we need to move in East and North directions. Otherwise, if $\frac{\Delta y_{c0}}{\Delta x_{c0}}$ of each node (x_c, y_c) is smaller than $(-\frac{HP}{HP+1})$, then we need to move in East and then in South direction. Also, when the Δx_{c0} is zero, if the Δy_{c0} is greater or equals zero then we need to move in East and North directions. Otherwise, we need to move in East and South directions. Once completed in which direction we have to move, we need to indicate how many bricks we need. We determine how many bricks are to be traversed East (E) and then North (N) to reach to the target Elevator by solving Equ 2 & 3.

$$(HP + 1)E + (HP)N = \Delta x_{c0} \quad \text{Equ 2}$$

$$-(HP)E + (HP + 1)N = \Delta y_{c0} \quad \text{Equ 3}$$

Similarly, by solving the following two equations (Equ 4 and 5) using two variables S and E, we can get the number of bricks to traverse in East (E) and then South (S) to directions the target Elevator.

$$(HP + 1)E - (HP)S = \Delta x_{c0} \quad \text{Equ 4}$$

$$-(HP)E - (HP + 1)S = \Delta y_{c0} \quad \text{Equ 5}$$

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOC

Pattern-based Placement and Assignment Algorithm

First Phase: Initialization

```
//(xc, yc) // Address of the current node
//(x0, y0) // Address of the reference Elevator
//(xte, yte) // Address of the temporary Elevator
Function East-North (Δxc0, Δyc0, HP)
```

```
{
//Solve "EN" Two Equivalents with two Variables!
// (HP + 1)E + (HP)N = Δxc0;
// -(HP)E + (HP + 1)N = Δyc0;

$$N = \left\lceil \left[ \frac{\Delta x_{c0}(HP) + \Delta y_{c0}(HP+1)}{(HP^2 + (HP+1)^2)} \right] \right\rceil$$
;

$$E = \left\lceil \left[ \frac{\Delta x_{c0} - HP \cdot (N)}{(HP+1)} \right] \right\rceil$$
;
(xt, yt) = (x0 + (N) · (HP), y0 + (N) · (HP + 1));
(xte, yte) = (xt + (E) · (HP + 1), yt - (E) · (HP));
}
```

Function East-South (Δx_{c0}, Δy_{c0}, HP)

```
{
//Solve "ES" Two Equivalents with two Variables!
// (HP + 1)E - (HP)S = Δxc0;
// -(HP)E - (HP + 1)S = Δyc0;

$$S = \left\lceil \left[ \frac{\Delta x_{c0}(HP) + \Delta y_{c0}(HP+1)}{-(HP^2 + (HP+1)^2)} \right] \right\rceil$$
;

$$E = \left\lceil \left[ \frac{\Delta x_{c0} + HP \cdot (S)}{(HP+1)} \right] \right\rceil$$
;
(xt, yt) = (x0 - (S) · (HP), y0 - (S) · (HP + 1));
(xte, yte) = (xt + (E) · (HP + 1), yt - (E) · (HP));
}
```

```
For all the nodes {
Δxc0 = xc - x0;
Δyc0 = yc - y0;
IF ((Δxc0 == 0) AND (Δyc0 ≥ 0)) THEN Call East-North;
ELSIF ((Δxc0 == 0) AND (Δyc0 < 0)) THEN Call East-South;
ELSIF ( $\frac{\Delta x_{c0}}{\Delta y_{c0}} \geq -\frac{HP}{HP+1}$ ) THEN Call East-North;
ELSIF ( $\frac{\Delta x_{c0}}{\Delta y_{c0}} < -\frac{HP}{HP+1}$ ) THEN Call East-South ;}
```

(a)

Second Phase: Post-Check

```
//(xte, yte) // Address of the temporary Elevator
//(xe, ye) // Address of the permanent Elevator
IF ((0 ≤ xte < M)) THEN
{
IF ((yte < 0) AND ((xte - (HP + 1)) ≥ 0)) THEN
//Go to the Elevator of West adjacent Region
(xe, ye) = (xte - (HP + 1), yte + (HP));
ELSIF ((yte < 0) AND ((xte + HP) < M)) THEN
//Go to the Elevator of North adjacent Region
(xe, ye) = (xte + (HP), yte + (HP + 1));
ELSIF ((yte ≥ N) AND ((xte + (HP + 1)) < M)) THEN
//Go to the Elevator of East adjacent Region
(xe, ye) = (xte + (HP + 1), yte - (HP));
ELSIF ((yte ≥ N) AND ((xte - HP) ≥ 0)) THEN
//Go to the Elevator of South adjacent Region
(xe, ye) = (xte - (HP), yte - (HP + 1));
ELSIF (0 ≤ yte < N) THEN
(xe, ye) = (xte, yte);
}
ELSE
{
IF ((xte < 0) AND ((yte - HP) ≥ 0)) THEN
//Go to the Elevator of East adjacent Region
(xe, ye) = (xte + (HP + 1), yte - (HP));
ELSIF ((xte < 0) AND ((yte + (HP + 1)) < N)) THEN
//Go to the Elevator of North adjacent Region
(xe, ye) = (xte + (HP), yte + (HP + 1));
ELSIF ((xte ≥ M) AND ((yte + HP) < N)) THEN
//Go to the Elevator of West adjacent Region
(xe, ye) = (xte - (HP + 1), yte + (HP));
ELSIF ((xte ≥ M) AND ((yte - (HP + 1)) ≥ 0)) THEN
//Go to the Elevator of South adjacent Region
(xe, ye) = (xte - (HP), yte - (HP + 1));
}
```

(b)

Figure 6.13: The pseudo code of the pattern-based placement and assignment algorithm

Figure 6.13 shows the pseudo code of the pattern-based placement and assignment algorithm. Figure 6.13.a shows the first phase and Figure 6.13.b presents the second phase of the algorithm. We have to mention that the rounded absolute value of E, noted $\lceil [E] \rceil$, and (N noted $\lceil [N] \rceil$ or S noted $\lceil [S] \rceil$) must be used in the algorithm. For example, if E equals 1.25, one East brick should be used and if E equals 1.64, two east bricks should be used. Using absolute of rounding values (E and (N or S)) leads to the nearest possible Elevator to the current node. As an example, the results of the pattern-based placement and assignment algorithm have been presented in Table 6.1 for a 5×5 Mesh layer when HP equals '1'. There are four exception nodes (see Table 6.1), their rows are highlighted and also they are shown in Figure 6.12 as well, for which their assigned Elevators have been placed out of the Mesh layer boundaries. Therefore in phase two (post-check), we have to answer to this question: are all the assigned Elevators within the Mesh boundaries? If the answer is no then we have to answer to this question: what is the nearest Elevator to the exception nodes, as hop-count is our main concern at this stage. Interestingly, due to the fact that the chosen out of boundary Elevators are only one hop far from the exception nodes (in X or in Y direction), their place can be only one of the four cases which are presented in Figure 6.13.b. There might be two possible nearest Elevators, therefore, if both conditions would be correct, only one of them should be selected. In such cases, the designer could decide based on the Region-Degree value of the Elevators which have been placed inside the Mesh layer or other design consideration.

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOC

Table 6.1: The results of the pattern-based placement and assignment algorithm for a 5×5 Mesh layer when HP equals ‘1’; Highlighted rows are exceptions for which the Elevators are out of the Mesh boundaries.

	Xc	Yc	Dx	Dy	DY/Dx	DRC	S	E	N	E	Xet	Yet	Xe	Ye
1	0	0	0	-2	DY<0	SE	1	0	NA	NA	-1	0	0	2
2	1	0	1	-2	-2	SE	1	1	NA	NA	1	-1	2	1
3	2	0	2	-2	-1	SE	0	1	NA	NA	2	1	2	1
4	3	0	3	-2	-0.667	SE	0	2	NA	NA	4	0	4	0
5	4	0	4	-2	-0.5	NE	NA	NA	0	2	4	0	4	0
6	0	1	0	-1	DY<0	SE	0	0	NA	NA	0	2	0	2
7	1	1	1	-1	-1	SE	0	1	NA	NA	2	1	2	1
8	2	1	2	-1	-0.5	NE	NA	NA	0	1	2	1	2	1
9	3	1	3	-1	-0.333	NE	NA	NA	0	1	2	1	2	1
10	4	1	4	-1	-0.25	NE	NA	NA	0	2	4	0	4	0
11	0	2	0	0	DY>0	NE	NA	NA	0	0	0	2	0	2
12	1	2	1	0	0	NE	NA	NA	0	0	0	2	0	2
13	2	2	2	0	0	NE	NA	NA	0	1	2	1	2	1
14	3	2	3	0	0	NE	NA	NA	1	1	3	3	3	3
15	4	2	4	0	0	NE	NA	NA	1	2	5	2	4	0
16	0	3	0	1	DY>0	NE	NA	NA	0	0	0	2	0	2
17	1	3	1	1	1	NE	NA	NA	1	0	1	4	1	4
18	2	3	2	1	0.5	NE	NA	NA	1	1	3	3	3	3
19	3	3	3	1	0.333	NE	NA	NA	1	1	3	3	3	3
20	4	3	4	1	0.25	NE	NA	NA	1	1	3	3	3	3
21	0	4	0	2	DY>0	NE	NA	NA	1	0	1	4	1	4
22	1	4	1	2	2	NE	NA	NA	1	0	1	4	1	4
23	2	4	2	2	1	NE	NA	NA	1	0	1	4	1	4
24	3	4	3	2	0.667	NE	NA	NA	1	1	3	3	3	3
25	4	4	4	2	0.5	NE	NA	NA	2	1	4	5	3	3

Figure 6.14 depicts three possible placements and assignments of Elevators in 5×5 Mesh layer. We simulate Figure 6.14.a, b, and c topologies with an analytical tool [71] to calculate the saturation threshold of the average latency. Then we compare these results with the best saturation threshold of the 18500 topologies with the same (N,E) sets. In Figure 6.14, the Region-Degree of each Region has been indicated next to its Elevator. Figure 6.14.a shows the pattern-based placement and assignment topology in which the exception nodes become Elevators to guarantee the maximum hop-count ‘1’ instead of post-check phase. Figure 6.14.b demonstrates the pattern-based placement and assignment topology with post-check phase in which the exception nodes have been assigned to the nearest possible Elevator in the layer.

The results show that although Figure 6.14.a has four more Elevators in each layer, its saturation threshold (8.52) is not very better than the Figure 6.14.b (7.14). Also, compared to the best saturation threshold among 18.5K topologies with nine Elevators in each layer, the saturation threshold of Figure 6.14.a is not close to or better than the best saturation threshold while Figure 6.14.b becomes saturated about the same point as the best of 18500 topologies with five Elevators in each layer. We believe, this can be justified with two main reasons:

Firstly, in Figure 6.14.a the Region-Degree of the exception nodes are one while the Region-Degree of other Regions are more than or equals to three. This means that the Elevators (vertical links) are not uniformly distributed between the nodes. In Figure 6.14.b, the saturation threshold is close to the best of 18.5K topologies, and although the maximum hop-count is not guaranteed to be one, the Region-Degrees values are close to each other. Secondly, in Figure 6.14.a the Regions with smaller Region-Degrees are placed in the edges, while the Regions with bigger Region-Degrees are placed in the center. This leads to

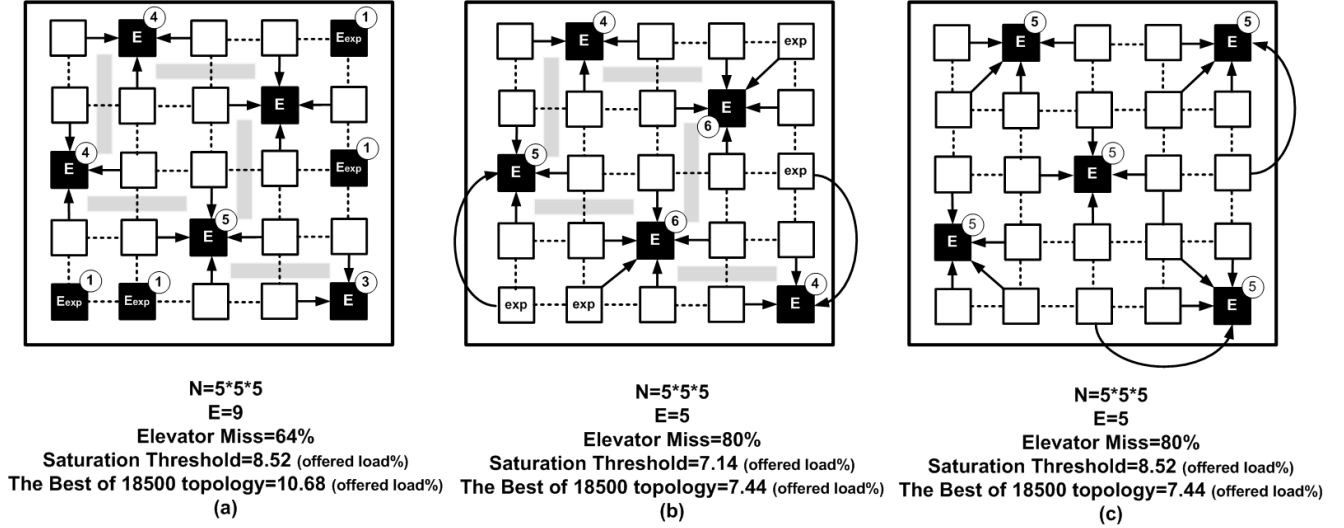


Figure 6.14: Three possible placement and assignment of Elevators in 5x5 Mesh layer

contention, specially in the central layers, because of the role of central layers as passages of the upper layers in Z- network and lower layers in Z+ network.

Consequently, not only the Region-Degrees of the Regions in the layers should be as similar as possible, but also the Elevators should be placed in the edges as much as possible, especially the Elevators (Regions) with bigger Region-Degree. Based on this remark, we placed and assigned five Elevators in each layer, as shown in Figure 6.14.c. Most of the Elevators are on the edges and the Region-Degree of all the Regions is five which equals $\frac{N}{E} = \frac{25}{5}$ where N is the layer size and E is the number of Elevators. This means that all the Elevators are assigned to the nodes uniformly and equally. This also proves a fact that hop-count is important when the Elevators are free to serve. Thus although pattern-based placement and assignment algorithm is a proper way to generalize the placement and assignment of the Elevators, it does not provide the best results. However, the designers can use the pattern-based methodology to define a pattern and generalize the placement and assignment. All in all, the conclusion of this section is this fact the Region-Degree is as important as the hop-count. Therefore, proposing a more general method to place the Elevators based on any network size and number of Elevators and assign them uniformly to the nodes would be practical and necessary.

4. Uniform Elevator Assignment (UEA)

In this thesis, the uniform traffic pattern is used due to our approach toward general purpose application platforms. Therefore, in order to provide a balance distribution of the Elevators between the 2D nodes, we propose a simple and practical method to set the Region-Degrees to the nearest values to average Region-Degree (RD_{avg}) which is calculated as Equ 6:

$$RD_{avg} = \frac{N}{E} \quad \text{Equ 6}$$

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOC

where N is the number of nodes in each layer and E is the number of Elevators. This means that in this method all the Elevators are assigned to approximately same number of 2D nodes in all the layers. This ends in a rough traffic balance in Elevators' buffers (FIFOs). In this general and uniform Elevators' assignment method, there are two types of Region-Degree which we call RD_1 and RD_2 . RD_1 is the ceiling value of average Region-Degree(RD_{avg}):

$$RD_1 = \left\lceil \frac{N}{E} \right\rceil \quad \text{Equ 7}$$

RD_2 is the floor value of average Region-Degree(RD_{avg}):

$$RD_2 = \left\lfloor \frac{N}{E} \right\rfloor \quad \text{Equ 8}$$

Moreover, NRD_1 and NRD_2 are the number of Regions that have Region-Degree RD_1 and RD_2 respectively:

$$NRD_1 = N - (RD_2 \times E) \quad \text{Equ 9}$$

$$NRD_2 = \frac{N - (NRD_1 \times RD_1)}{RD_2} \quad \text{Equ 10}$$

4.1. Experimental Results of Uniform Elevator Assignment Method

Since placement also has an important role in Vertically-Partially-Connected 3D-NOc, we evaluate the performance of the Uniform Elevator Assignment method based on two different placement strategies:

1. The rule of the first placement strategy (UEA_HC in Figure 6.15):
 - a. The Elevators can be adjusted in any place (address) if they connect to their 2D nodes with the least possible hop-count(s).
2. The rules of the second placement strategy (UEA_Edge in Figure 6.15):
 1. Placement of the Elevators should be on the edges and specially on the corners of a Mesh layer, while their assigned 2D nodes should be selected from the hot-spot area in the center of Mesh layer.
 2. Placement of the Elevators in same networks (ex: UP network) in adjacent Mesh layers should be different.
 3. The placement of the Elevators in different networks (UP and DOWN networks) in a Mesh layer should be different.
 4. X-First routing paths from the 2D nodes to their Elevators in different networks (UP and DOWN networks) in a Mesh layer should be different and not overlapped as much as possible.

Based on the Uniform Elevator Assignment method two samples were manually generated based on the described first and second placement strategies for different number of Elevators. In the chosen topologies, the number of nodes (N) is 25 (5×5), the number of

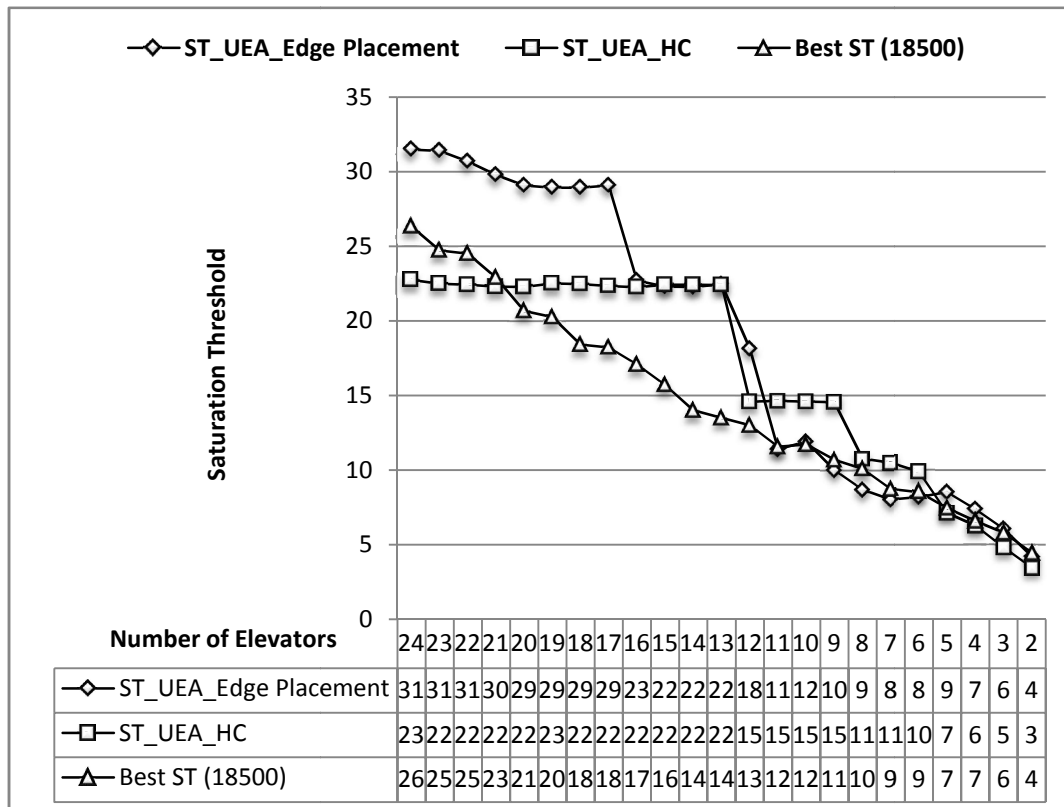


Figure 6.15: The comparison between the Uniform Elevator Assignment (UEA) and random topologies (placements and assignment) in (5×5×5) network

layers (Z) is 5. Also, to compare the results of the Uniform Elevator Assignment topologies, we use the saturation threshold values of the best of 18.5K topologies with the same (N), (E), and (Z) which have been randomly (Random Placement and Random Assignment) generated (see Figure 6.15).

As can be seen in Figure 6.15, when the number of Elevators (E) is near to the number of nodes in each layer (N) and '0', the Uniform Elevator Assignment samples based on the second placement strategy outperforms compared to the first one and the best topology of 18.5K samples. When (E) is near to (N), most of the nodes are Elevators and if we chose a 2D node from hot-spot area which sends its packets to the edges or corners, not only we reduce the arbitration latency of the 2D router in hot-spot, but also forward the extra load of Elevator First routing to the areas (edge and corner) with less links and buffers utilization and to the Elevators with less number of 2D ports. On the other hand when (E) is near to '0', most of the nodes do not have vertical links and have already been assigned to small number of Elevators. Therefore, few of Elevators have to serve a large number of 2D nodes. In this situation, it does not really matter how many hops 2D nodes' packets should traverse to reach to their Elevators. However, the placement of busy Elevators on the edges and corners with less traffic can slightly improve the performance.

When the number of Elevators (E) is near and more than $(\frac{N}{2})$, the first and second strategies performs almost similarly, and of course they have better saturation threshold than the best topology of 18.5K samples. While, when the number of Elevators (E) is reduced to less than $(\frac{N}{2})$, the Uniform Elevator Assignment samples based on the first placement strategy outperforms the other ones. This means that hop-count plays an important role when (E) is less than $(\frac{N}{2})$ and not near to '0' in Vertically-Partially-Connected 3D-NoC. In such case, the number of Elevators is less than the number of 2D nodes. So, if we place all the Elevators on the edges and corners and the 2D nodes on the center then more than $(\frac{N}{2})$ of nodes want to send their packets to the edges and corners. This not only does not reduce the Elevator First traffic in a Mesh layer, but also increases the traffic load in hot-spot area. This is why distributed placement of Elevators as near to their assigned 2D nodes can be a proper way to save the performance when (E) is less than $(\frac{N}{2})$ and not near to '0'.

Also, the saturation thresholds of Uniform Elevator Assignment samples based on the first placement strategy are almost similar from (E) equals 24 to (E) equals 13. It means that by the reduction of 12 Elevators in each layer (48 in the network) the performance can be saved. In addition, the saturation thresholds of Uniform Elevator Assignment samples based on the second placement strategy are almost similar from (E) equals 24 to (E) equals 17 which shows by the reduction of 8 Elevators in each layer (32 in the network) the performance can be saved. Consequently, it means that the the Uniform Elevator Assignments topologies can tolerate more network loads with less number of Elevators (vertical links).

Finally, it must be mentioned that although these results are interesting, due to the huge number of possible topologies we really can not know the best configuration. However, we can say that Uniform Elevator assignment method can obtain network with saturation thresholds above the average.

5. Quadratic-Based Estimation Method

In this section, we present a quadratic-based estimation method. We believe that, since the performance of Vertically-Partially-Connected 3D-NoC is influenced by different kinds of architectural parameters, which have already been described in section 2, mutual multiplication of the parameters in a quadratic form can predict the behavior of Vertically-Partially-Connected 3D-NoC architectures. The quadratic form is calculated as follows:

$$(P_1\beta_1 + P_2\beta_2 + \dots + P_n\beta_n) + (P_1P_2\beta_{n+1} + P_1P_3\beta_{n+2} + \dots + P_{n-1}P_n\beta_M) = Y \quad \text{Equ 11}$$

where P_i is the i th parameter and β_i is the coefficient of P_i parameter. Also P_iP_j is the multiplication of both P_i and P_j parameters. The number of coefficients M , which is more than the number of parameters n , is calculated as:

$$M = \sum_{i=1}^n i \quad \text{Equ 12}$$

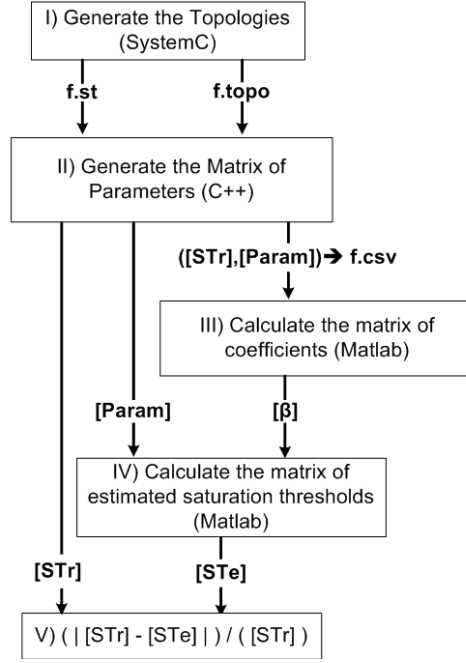


Figure 6.16: The flow of quadratic-based estimation method

Finally for S samples (topologies), the estimated saturation threshold $[ST]$ is calculated as follows:

$$\underbrace{\begin{bmatrix} P_{11}, P_{12}, \dots, P_{1n}, P_{11}P_{12}, P_{11}P_{13}, \dots, P_{1n-1}P_{1n} \\ P_{21}, P_{22}, \dots, P_{2n}, P_{21}P_{22}, P_{21}P_{23}, \dots, P_{2n-1}P_{2n} \\ \vdots \\ P_{S1}, P_{S2}, \dots, P_{Sn}, P_{S1}P_{S2}, P_{S1}P_{S3}, \dots, P_{Sn-1}P_{Sn} \end{bmatrix}}_{[Param]_{S \times M}} \times \underbrace{\begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}}_{[\beta]_{M \times 1}} = \underbrace{\begin{bmatrix} ST_1 \\ ST_2 \\ \vdots \\ ST_S \end{bmatrix}}_{[ST]_{S \times 1}} \quad \text{Equ 13}$$

Therefore, the flow for saturation threshold estimation based on the quadratic method is depicted in Figure 6.16, which contains 4 main steps:

1) Generate the samples (topologies): in this step, random topologies ($f_i.topo$) of Vertically-Partially-Connected 3D-NoC with different and random placement and assignment are generated. Also the saturation thresholds ($f_i.st$) of the generated topologies under Elevator First routing algorithm (Chapter Four) [68] and Elevator First routers (Chapter Five) [72] are calculated by the analytical method which is presented in [71].

2) Generate the matrix of parameters: in this step, the matrix of parameters $[Param]$ is produced by parsing the generated topologies.

3) Calculate the matrix of coefficients (Method Learn phase): in this step the matrix of coefficients $[\beta]$ is calculated as Equ 14 based on both matrix of parameters $[Param]$ and matrix of saturation threshold $[ST_r]$:

$$[\beta] = ([Param]^t \cdot [Param])^{-1} \cdot [Param]^t \cdot [ST_r] \quad \text{Equ 14}$$

4) Calculate the matrix of estimated saturation thresholds (Method Test phase): in this step the estimated saturation threshold is calculated as follows using Equ 13 and Equ 14:

$$[Param]_{S \times M} \times [\beta]_{M \times 1} = [ST_e]_{S \times 1} \quad \text{Equ 15}$$

5.1. The Parameters in Matrix of Parameters [Param]

Based on the described methodology and the four Region-based parameters, we define the six main architectural parameters as follows, which are used in the matrix of parameters [Param] and shown in Table 6.2:

Network Average hop-count(AHC_{uni}): The average hop-count of a topology is different for different traffic patterns and routing algorithms. Therefore, for a specific traffic pattern and routing algorithm, this parameter shows the effect of topology and routing algorithm on the average latency and power consumption of a NoC. Since in this chapter the traffic pattern is uniform, the network average hop-count is the average of the Manhattan hops between all the nodes of the topology under Elevator First routing algorithm. This parameter is both under influence of placement of Elevators and assignments of 2D nodes.

Standard deviation of Region-Degree(RD_{std}): We showed previously that if the Region-Degrees of Elevators are set to the values which are close to the average of Region-Degree ($\frac{N}{E}$), the topologies under uniform traffic have a high saturation threshold. Therefore, we consider the standard deviation of Region-Degree in the network to observe whether the Region-Degree of the Regions are close to the average value or not and also its effect on the saturation threshold. This parameter is just under influence of the assignments of 2D nodes.

Standard deviation of Regions Hop-Count($RAHC_{std}$): This depends on both placement and assignment parameters. If we assume that the ideal Region-Degree for all the Regions in a layer is the average of Region-Degree (RD_{avg}), then the ideal Region Hop-Count of such a Region is calculated as Equ 16:

$$RHC_{idl} = \frac{-2([h]^3 + 3[h]^2 + (\frac{N}{E} - 3)[h] + (\frac{N}{E} - 1))}{3(\frac{N}{E} - 1)} \quad \text{Equ 16}$$

where h is the hop-count between each node and its Elevator in the Region's Elevator and calculated as Equ 17:

$$h = \frac{-1 + \sqrt{1 + 2(\frac{N}{E} - 1)}}{2} \quad \text{Equ 17}$$

Appendix A details how these equations are obtained.

Standard deviation of average Elevators distances(ED_{std}): We consider this parameter to observe the distribution of Elevators in each layer. Therefore, this is a placement parameter. The average Elevators distances in all the layers are calculated as Equ.18:

$$ED_{avg} = \frac{\sum_{k=1}^Z \sum_{i=1}^{R_k} \sum_{j=1}^{R_k} (|x_{elvki} - x_{elvkj}| + |y_{elvki} - y_{elvkj}|)}{R_t \times Z} \quad \text{Equ 18}$$

where R_t is the number of regions in each layers and Z is the number of layers. The standard deviation of average Elevators distances shows how the Elevators' placements in the different layers are similar to each other.

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NOC

Table 6.2: The architectural parameters of Vertically-Partially-Connected 3D-NoC
(Z is the number of layers, N is the number of nodes in each layer, E is the number of Elevators in each layer, and R_t is the number of Regions in each layer)

#	Parameter	Type		Formula
		Placement	Assignment	
1	Network's Average hop-count	✗	✗	The average hop-count of the Elevator First Routing algorithm in vertically partially-connected 3D NoC
2	Standard deviation of Region-Degree	-	✗	$RD_{std} = \sqrt{\frac{\sum_{k=1}^Z \sum_{j=1}^{R_k} (RD_{jk} - RD_{avg})^2}{E \times Z}}$
3	Standard deviation of Regions Hop-Count	✗	✗	$RAHC_{std} = \sqrt{\frac{\sum_{k=1}^Z \sum_{j=1}^{R_k} \sum_{i=1}^{M_j} (RHC_{ijk} - RHC_{idl})^2}{N \times Z}}$
4	Standard deviation of average Elevators distances	✗	-	$ED_{std} = \sqrt{\frac{\sum_{k=1}^Z (ED_k - ED_{avg})^2}{Z}}$
5	Standard deviation of Load-Region-Degree	✗	✗	$LRD_{std} = \sqrt{\frac{\sum_{k=1}^Z \sum_{j=1}^{R_k} (L_{ki} \cdot (RD_{jk} - RD_{avg}))^2}{R_t}}$
6	Standard deviation of Total-Region-Degree	✗	✗	$TRD_{std} = \sqrt{\frac{\sum_{k=1}^Z (Z - k) \cdot \sum_{j=1}^{R_k} (TRD_{jk} - (k \cdot RD_{avg}))^2}{R_t}}$

Standard deviation of Load-Region-Degree (LRD_{std}): Region-Degree is an important parameter which roughly can show the load of the Region. If the Region-Degree of a Region is a big number, it means that the Region's Elevator have to serve a higher number of packets. On the other hand, the loads of the Region members are quite important. If the Region-Degree of a Region is a big number but the members do not inject considerable amount of packets to the network, the big value of Region-Degree should not be considered. In order to address this issue we define Standard deviation of Load-Region-Degree. In the formula for this parameter, the Region-Load of each Region is multiplied by the difference of the Region-Degree of Region and the average of the Region-Degrees, which acts like a coefficient to provide a more realistic view of the Regions' performance of the topology.

Standard deviation of Total Region-Degree (TRD_{std}): As described before, in Vertically-Partially-Connected 3D-NoC the layers put extra loads on each other. Based on the functionality of Elevator First routing these loads reduce when the Elevators are placed similarly in all the layers, since the packets do not need extra hops from the receiver node to their Elevators in intermediate layers to be forwarded to their destination layers. We thus define TRD_{std} , a parameter which is influenced by both placement and assignment of Elevators.

5.2. Experimental Results of Quadratic-based Estimation Method

In this section, we present the experimental results of quadratic-based estimation method. From a practical point of view, to determine the β_i in Equ 13, the randomly generated topologies and their saturation thresholds are parsed to produce an input file for Matlab, which is used as to solve the equation. Firstly, we use the 18500 samples of topology to make the matrix of parameters [Param] and their saturation threshold under uniform traffic pattern to make the matrix of saturation threshold [ST] for the different values of Elevators in network size equals $(6 \times 6 \times 5)$, $(5 \times 5 \times 5)$, $(4 \times 4 \times 5)$, and $(3 \times 3 \times 5)$. Actually the $(\log_{10} 18500)$ equals 4.26 which compared to the number of possibilities given in Figure 6.3

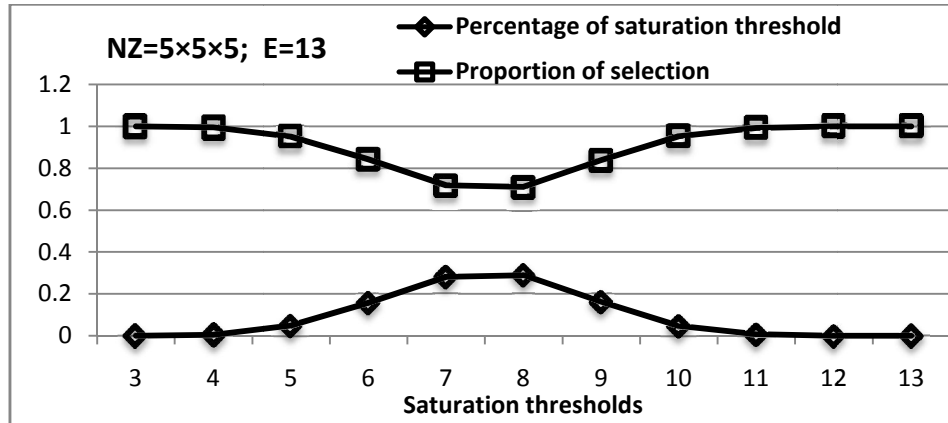


Figure 6.17: The proportion of prioritized-based selection

is epsilon. As mentioned before, the number of nodes and Elevators are similar in all the layers and the number of layers Z is 5 constantly. We consider Z constant because it affects neither placement nor assignment.

As described before, the quadratic-based estimation method has two main phases: a) learn the method by calculation of the matrix of coefficient $[\beta]$ and b) test the method by calculation of estimated Y, which is the matrix of estimated saturation point $[ST_e]$. Therefore, the samples have to be divided into two sets, the learning set and the testing one. Fundamentally, more samples in learning set make the $[\beta]$ matrix more reliable and reduce the average error. However it is important to select the samples intelligently. For this reason, we examine the samples with two methods: 1) Random-based sample selection, 2) Prioritized-based sample selection. In Random-based sample selection 16000 samples are randomly chosen out 18500 ones to learn the method. In Prioritized-based sample selection, the samples are chosen based on the percentage of presence of their saturation threshold. We first sort the samples based on their saturation thresholds and calculate the percentage of their presences. Then we choose the learning samples based on the proportion of their saturation threshold:

$$\text{Proportion} = 1 - \frac{STQ}{MS} \quad \text{Equ 19}$$

where the STQ is the saturation threshold quantity in the main set and MS is the total number of samples in the main set. This means that when the quantity of a saturation threshold is big the probability of being chosen is small compared to the others. This ends in a smooth selection of samples among all the saturation thresholds. Figure 6.17 depicts this concept for a sample set of network size equals $(5 \times 5 \times 5)$ and $E = 13$.

Finally, in Figure 6.18, 6.19, 6.20, and 6.21 the average errors and standard deviation of average errors for network size equals $(6 \times 6 \times 5)$, $(5 \times 5 \times 5)$, $(4 \times 4 \times 5)$, and $(3 \times 3 \times 5)$ are presented. As can be seen, in all network sizes the average errors are improved (constantly about 10%) by the reduction of number of Elevators (when 75% of nodes are

Elevators). Also the average errors of Prioritized-based sample selection method are about 5% better than the average errors of Random-based sample selection method.

When the number of Elevators (E) is close to the number of nodes (N), the Regions are small. The Regions have at most one assigned 2D node to their Elevators. Most of the Region-Degrees are '1' which means the Region contains just an Elevator and no 2D nodes. Thus Region hop-count is zero. Therefore the four defined Region's parameters are not very different in different topologies. On the other hand, when the number of Elevators is reduced, the number of 2D nodes is increased. This leads to Regions with more 2D nodes assigned to Elevators. Consequently, the four described Region parameters become bigger and more different from one topology to another. Using this not only makes the characteristics and behavior of the network appear clearly but also the variety in input parameters helps the quadratic method to estimate the saturation threshold based on more distributed inputs. This is why we believe the average error of saturation threshold is reduced in both random and prioritized sample selection methods. However, in prioritized sample selection method, the samples are selected based on the percentage of presence of the possible saturation thresholds. Therefore (β) matrix is made based on the saturation threshold which are more probable than the others. This is the reason why the average error prioritized sample selection method is about 5% less than the random one.

However, the standard deviation of the errors is not negligible (about 0.07) for both random and Prioritized-based sample selection methods. Although this method can roughly estimate the average of saturation threshold for each (N, E) set, the described fact shows that this method needs to be improved. This can be done by two main actions: 1) find a set of parameters which are more correlated with each other and 2) define more architectural parameters which are more appropriate and can show a better view of the performance of Vertically-Partially-Connected 3D-NoC. These are the main open questions and the scope of a future work.

6. Conclusion

In this chapter, we firstly define and study the parameters which affect the performance of Vertically-Partially-Connected 3D-NoC. Secondly, we discuss and evaluate a placement and assignment method which guarantees minimum possible hop-count between the 2D nodes and their Elevators and uniformly placed the Elevators through in the layers. Thirdly we equally assign the 2D nodes to the Elevators, and finally present an estimation quadratic-based method to estimate the saturation threshold of Vertically-Partially-Connected 3D-NoC.

To summarize, we showed that the pattern-based placement and assignment algorithm can be generalized and customized based on the patterns that designers define. Also, the results showed that if the Elevators are placed in the layers and assigned to the nodes uniformly, the results will be better. Also placement of the Elevators on the edges can forward the Z+ and Z- traffic to the edges and alleviate the load of hot spot. Moreover, the topologies based on the uniform Elevator assignment strategy have reasonably good performance which shows that the average Region-Degree like hop-count is one the key

CHAPTER SIX:

ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NoC

parameters in the performance of Vertically-Partially-Connected 3D-NoC. Finally the results of the quadratic-based estimation method show that although the standard deviation of the error is high (about 0.07), the method can be a potential solution to estimate properly the saturation threshold of the Vertically-Partially-Connected 3D-NoC topologies.

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NoC

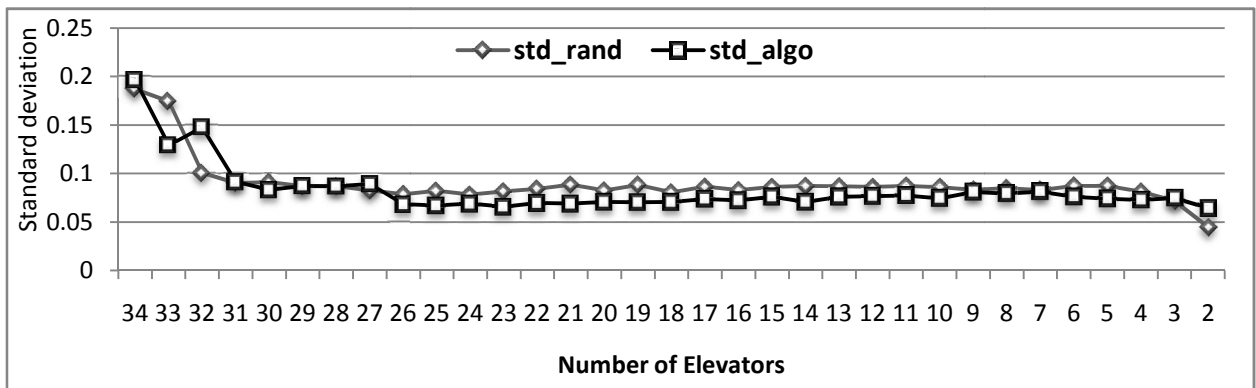
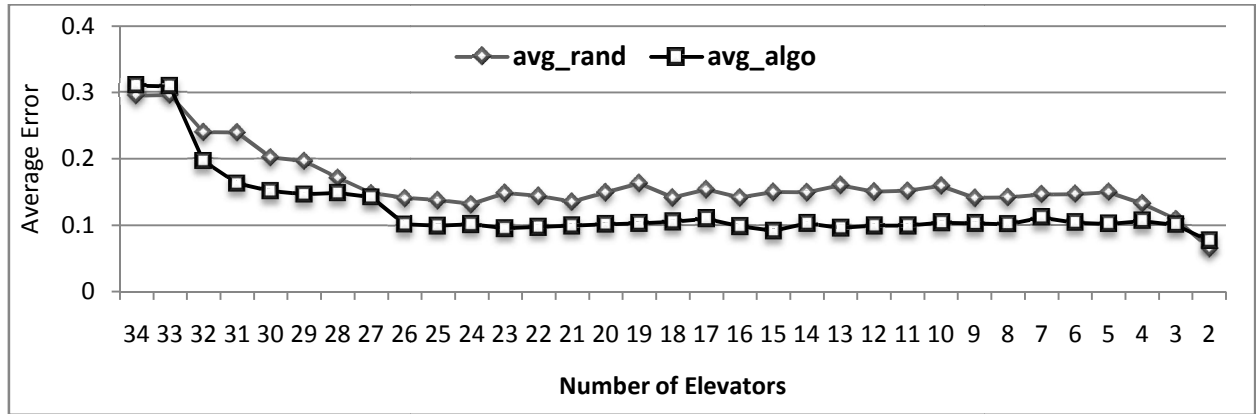


Figure 6.18: Average and standard deviation of errors when $NZ = 6 \times 6 \times 5$

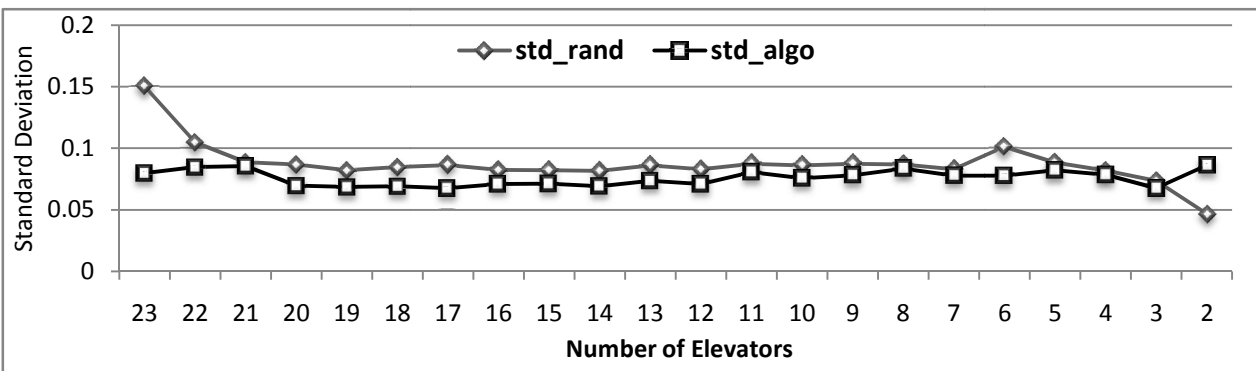
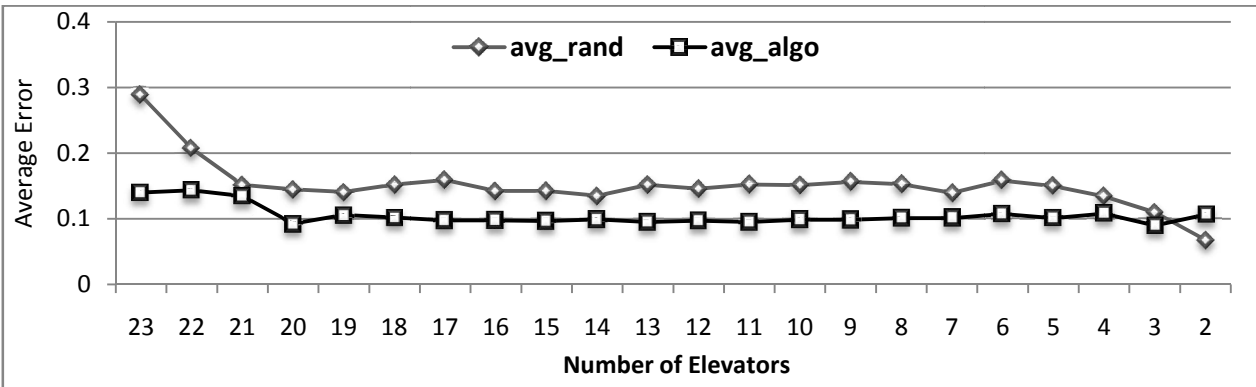


Figure 6.19: Average and standard deviation of errors when $NZ = 5 \times 5 \times 5$

CHAPTER SIX: ARCHITECTURAL EXPLORATION OF VERTICALLY-PARTIALLY-CONNECTED 3D-NoC

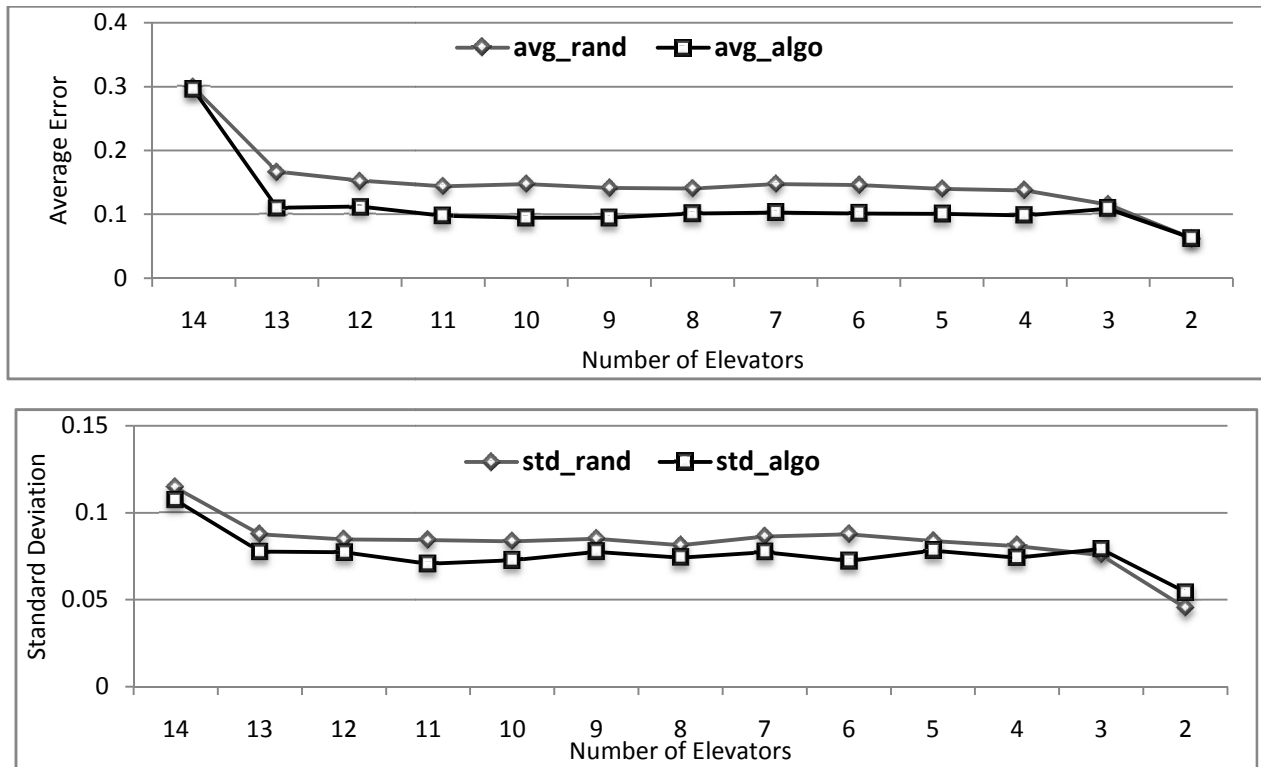


Figure 6.20: Average and standard deviation of errors when $NZ = 4 \times 4 \times 5$

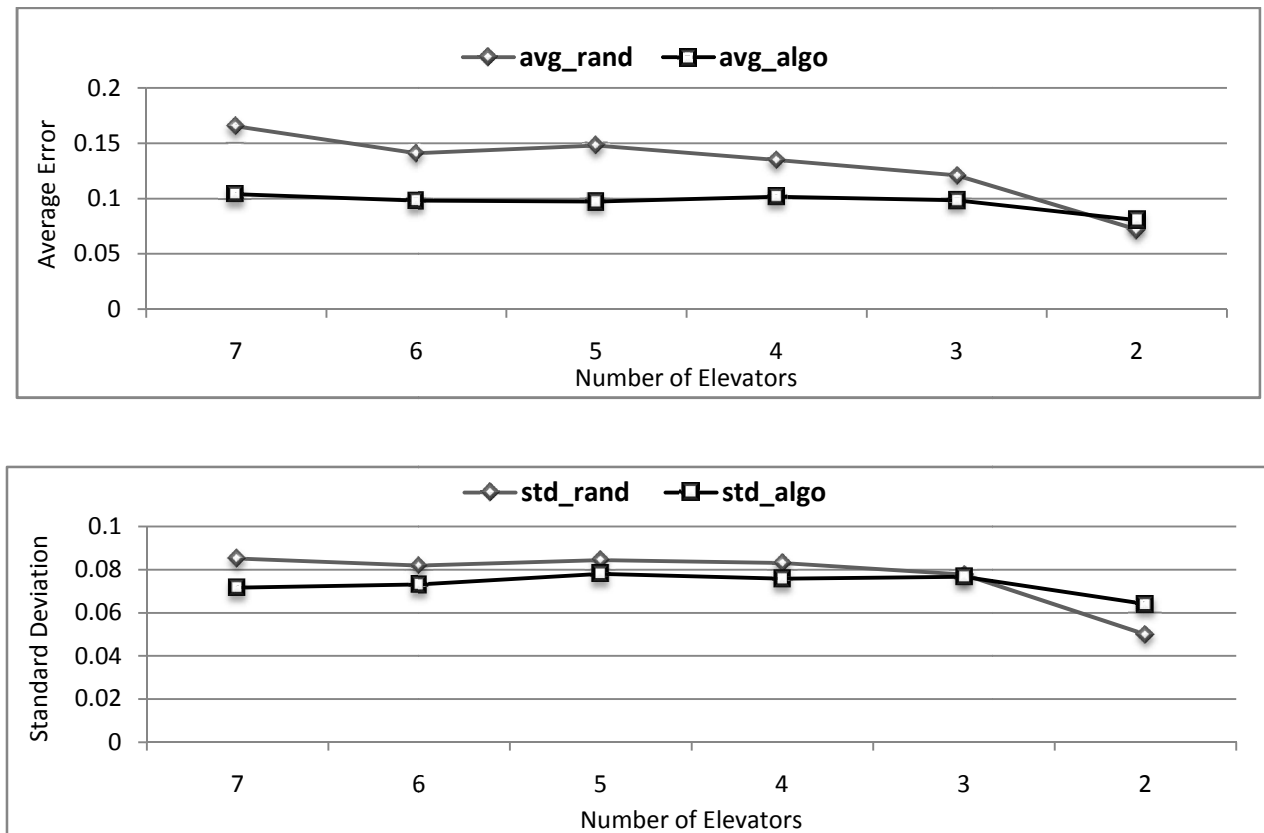


Figure 6.21: Average and standard deviation of errors when $NZ = 3 \times 3 \times 5$

CHAPTER SEVEN: CONCLUSION AND FUTURE WORKS

To overcome the yield and flexibility issues in 3D NoC architectures introduced in Chapter Two (Problem Definition), we presented the concept of Vertically-Partially-Connected 3D-NoC architectures. This architecture is a stacked NoC-based SoC in which each stacked layer is not fully connected to each adjacent layer. Therefore, in such architecture some of the routers do not have UP and/or DOWN vertical ports. We also discussed about challenges of Vertically-Partially-Connected 3D-NoC which are 1) Routing and 2) Performance Degradation. In addition, we explicated that performance in NoC is influenced by both micro-architectural (Router) and architectural (Topology) parameters and elements. Consequently, we posed three main questions that we tried to address in this thesis:

First: Routing in Vertically-Partially-Connected 3D-NoC:

Question:

If a node (router) does not have any UP port and needs to send a packet to an upper layer, which path should be selected? Also, if a routing algorithm is selected to determine a routing path in the case, is it deadlock-free?

Answer:

We answered to this question in Chapter Four by the explanation and evaluation of a deadlock-free routing algorithm called Elevator First. When a router does not have UP port and needs to send a packet to upper layers, Elevator First routing forwards the packet to the immediate upper layer by using an intermediate node called Elevator that has an UP port. A temporary header containing the Elevator's address is added to the packet and then it is forwarded to the Elevator by X-First Deadlock-free routing. To cope with the deadlock issue in Elevator First routing algorithm, the flow of packets goes to upper layers is completely and physically separated from the flow of packets goes to lower layers. Therefore, two networks have been defined called Z+ (contains the packets going to upper layers) and Z- (contains the packets going to lower layers). This assures that a cycle never happens.

A Cycle-Accurate SystemC model of the Elevator First router has been developed to study the practicality of the Elevator First routing algorithm. We analyzed the results of the average latencies of a $5 \times 5 \times 5$ network versus the flit injection rate (offered load of cores, as

the fraction of maximum possible load, i.e. one flit per cycle) in different situations when respectively using uniform traffic distribution (in which each core sends packets uniformly to all other cores with a same probability and a same rate) and localized one (when closest destinations of being reached have a higher probability). Finally we observed that by reduction of the number of 3D nodes in Vertically-Partially-Connected 3D-NoC, the saturation threshold of the network's average latency increases. This is more serious in Uniform traffic distribution than in Localized one. However, when 10% of vertical links are removed the network performance under localized traffic is approximately the same as that of a fully-connected 3D-Mesh network using the usual Z-First routing algorithm. Also, Elevator First routing has been compared with Z-First routing algorithm in a fully-connected 3D NoC with the same buffer size. As we expected, the Elevator First routing has better saturation threshold, due to the separation of UP and DOWN traffics via Z+ and Z- networks respectively.

Second: Performance Degradation (Router Micro-Architectural):

Question:

As router micro-architecture plays a key role in network latency, what is a tradeoff between the advantages of router architecture and its costs? How a router in Vertically-Partially-Connected 3D-NoC can be designed to be as simple as possible, to add the least load, and to handle the Elevator First routing mechanism?

Answer:

We addressed these issues in Chapter Five by proposing a router micro-architecture based on Elevator First routing algorithm. In addition, we have developed Elevator-First 3D router in VHDL and synthesized it with Synopsis using CMOS 65nm technology of STMicroelectronics (GP65LVT). We synthesized the router with 0ns constraint to achieve the maximum operational frequency. In Chapter Five, we showed that the area overhead of Elevator First router, compared to an equivalent 7 ports router with the same buffer size, is only 8%. This overhead is mainly due to the Elevator First FSMs in all the input ports, an extra routing module in all 2D input ports, an extra arbitration module in all 2D output ports, and the multiplexers which separate the Z+ and Z- networks.

Third: Performance Degradation (System-Topology):

Question:

The mixture of a specific number and placement of Elevators (vertical links) and their assignment to the 2D nodes creates a topology in Vertically-Partially-Connected Mesh-based 3D-NoC. As by reduction of the number of Elevators, saturation threshold of average latency increases, how can the designer properly place and assign the Elevators to prevent or at least limit performance degradation? Furthermore, are there any topology related parameters that model accurately the behavior of the Vertically-Partially-Connected 3D-NoC? If there are, how can they show the changes in performance and why? Nevertheless, the huge number of topology possibilities is a serious problem which can easily threat the evaluation of architectural exploration in Vertically-Partially-Connected 3D-NoC. Consequently, is there

any simple and practical method which quickly can estimate the saturation threshold of average latency as a performance indicator?

Answer:

We addressed these questions in Chapter Six. First of all, based on the concept and structure of Vertically-Partially-Connected 3D-NoC, we defined some architectural parameters such as the number of the nodes an Elevator should serve and the average hop-counts between the Elevator and the assigned nodes. Then we presented a placement method based on predefined patterns to place similar patterns on a layer. In the “pattern-based placement” method, the Elevators are placed on each layer uniformly. As an example, we showed a pattern with minimum hops between each Elevator and its assigned nodes for $5 \times 5 \times 5$ network.

Experiments of the Pattern-based placement topologies showed that when the Region-Degrees of all the Regions are approximately similar in the network, the network has better performance compared to the network with different Regions’ Region-Degrees. Therefore, we used average Region-Degree ($\frac{N}{E}$) to present a method to assign the nodes to the Elevators equally and uniformly called “Uniform Elevator Assignment”. We evaluated this method by two different placement strategies based on 1) “hop-count placement” which guarantees minimum hop-count between the nodes and their Elevators and 2) “Elevator corner and edge placement” which tries to place the Elevators on corners and edges and provide non overlapping Z+ and Z- networks. The results show that when E is more than average Region-Degree ($\frac{N}{E}$), the second placement strategy is better than the first one. While when E is less than average Region-Degree ($\frac{N}{E}$), the first placement strategy is better than the second one. Therefore, we can conclude that not only all the defined parameters are useful, but also devising an algorithm which can place and assign the Elevators based on the conclusions automatically taking only mesh size and number of Elevators as inputs would be very practical. Also, for more reliable comparison it is better to compare the results of the algorithm with more randomly generated samples. This is one of the future works of this thesis.

Finally we used the standard deviation of defined parameters to propose a quadratic-based method to estimate the saturation threshold of Vertically-Partially-Connected 3D-NoC. By analyzing the experimental results, we observe that the average error of the method is about 5%. However, the standard deviation of the errors is not negligible (about 0.07). Therefore, although this method can roughly estimate the average of saturation threshold for each (N, E) set, the described fact shows that this method needs to be improved. This can be done by two main actions: 1) find a set of parameters which are more correlated with each other and 2) define more architectural parameters which are more appropriate and can show a better view of the performance of Vertically-Partially-Connected Mesh-based 3D-NoC. These are other future works of this thesis.

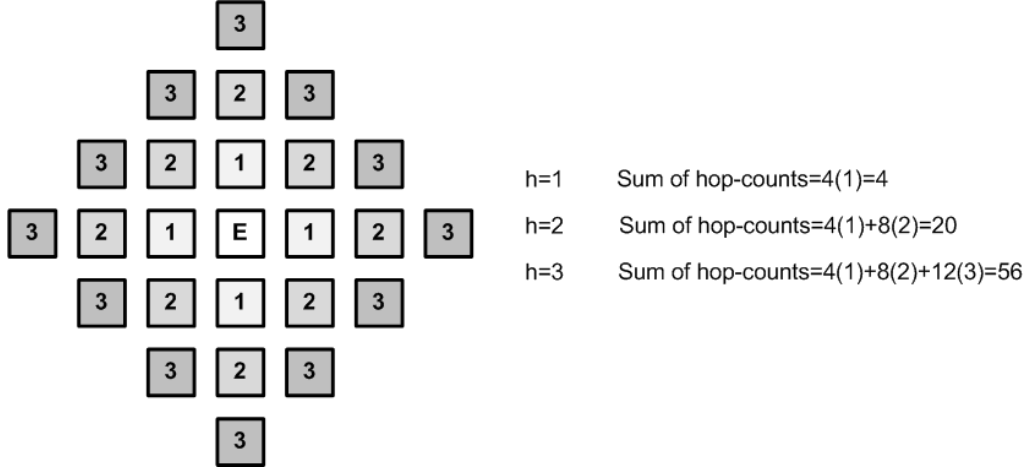


Figure A.1: B is the sum of hop-count between A number of 2D nodes and their Elevator

Appendix A: Ideal Region Hop-Count

In this thesis, we calculated Ideal Region-Degree (IRD) as follows:

$$IRD = RD_{avg} = \frac{N}{E} \quad \text{Equ 1}$$

Also, we define (A) as the maximum number of 2D nodes which can be assigned to an Elevator by hop-count h which is calculated as follows:

$$A = \sum_{i=0}^h 4i \quad \text{Equ 2}$$

We define (B) as the sum of hop-counts between A number of 2D nodes and their Elevator which is calculated as follows (see Figure A.1).

$$B = \sum_{i=0}^h 4i^2 \quad \text{Equ 3}$$

Therefore, based on A, we define the ideal Number of 2D nodes which can be assigned to an Elevator. This is a number between the maximum number of 2D nodes which can be assigned to an Elevator by hop-count (h) and (h + 1):

$$\left(\sum_{i=0}^h 4i\right) \leq \left(\frac{N}{E} - 1\right) < \left(\sum_{i=0}^{h+1} 4i\right) \quad \text{Equ 4}$$

The simple form of Equ 4 is shown in Equ 5:

$$4 \left(\frac{h \cdot (h+1)}{2}\right) \leq \left(\frac{N}{E} - 1\right) < 4 \left(\frac{(h+1) \cdot ((h+1)+1)}{2}\right) \quad \text{Equ 5}$$

By solving the Equ 5, we can define the Ideal hop-count between all the 2D nodes and an Elevator as follows:

$$h = \frac{-1 + \sqrt{1 + 2\left(\frac{N}{E} - 1\right)}}{2} \quad \text{Equ 6}$$

Therefore, based on the Equ 5 and 6, the ideal Region Hop-Count (RHC_{idl}), which is the average of hop-counts between all the Region members and their Elevator in an ideal Region, can be calculated as follows:

$$RHC_{idl} = \frac{\sum_{i=0}^{IRD} HC_i}{IRD} = \frac{\left(\sum_{i=0}^{\lfloor h \rfloor} 4i^2\right) + \left(\left(\left(\frac{N}{E} - 1\right) - \sum_{i=0}^{\lfloor h \rfloor} 4i\right) \times (\lfloor h \rfloor + 1)\right)}{\frac{N}{E}} \quad \text{Equ 6}$$

$$RHC_{idl} = \frac{4 \left(\frac{\lfloor h \rfloor (\lfloor h \rfloor + 1) (2\lfloor h \rfloor + 1)}{6}\right) + \left(\left(\left(\frac{N}{E} - 1\right) - 4 \left(\frac{\lfloor h \rfloor (\lfloor h \rfloor + 1)}{2}\right)\right) \times (\lfloor h \rfloor + 1)\right)}{\frac{N}{E}} \quad \text{Equ 7}$$

$$RHC_{idl} = \frac{-2(\lfloor h \rfloor^3 + 3\lfloor h \rfloor^2 + \left(\frac{N}{E} - 3\right)\lfloor h \rfloor + \left(\frac{N}{E} - 1\right))}{3\left(\frac{N}{E} - 1\right)} \quad \text{Equ 8}$$

ACRONYM

Acronym

SoC	System-on-Chip
IP	Intellectual Property
NoC	Network-on-Chip
3D NoC	Three Dimensional Network-on-Chip
TSV	Through-Silicon-Vias
VL	Vertical link
N	Number of nodes in a layer
Z	Number of layers
E	Number of Elevator
Elv_u	UP_Elevator
Elv_d	DOWN_Elevator
Elv_u/d	UP/DOWN_Elevator
EF	Elevator First
VN	Virtual network
ST	Saturation Threshold
Uni	Uniform Distribution Traffic
Loc	Localized Distribution Traffic
SF	Switch Fabric
BoP	Beginning of Packet
EoP	End of Packet
ROK	Read is OK
WOK	Write is OK
ADRC	Current Address
ADDR	Destination Address
OT	Ordinary Transmission
AH	Header Addition
RH	Header Removal
Req	Request
Rdy	Ready
FSM	Finite State Machine
RR FSM	Round Robin FSM
P	Placement
A	Assignment
T	Topology
RAHC	Regions average Hop-Count
RD	Region-Degree
LRD	Load-Region-Degree
TRD	Total Region-Degree
HP	Hop-Count
UEA	Uniform Elevator Assignment
[Param]	Matrix of parameters
[β]	Matrix of coefficients
STQ	Saturation threshold's quantity

REFERENCES

References

-
- [1] K. Olukotun, B. Nayfeh, L. Hammond, K. Wilson, and K. Y. Chang, "The Case for a Single-Chip Multiprocessor," in Proceedings of the 7th International Symposium on Architectural Support for Programming Languages and Operating Systems (ASPLOS) 1996.
- [2] Chrysostomos Nicopoulos, Vijaykrishnan Narayanan, Chita R. Das: Network-on-Chip Architectures - A Holistic Design Exploration. Lecture Notes in Electrical Engineering 45, Springer 2010, ISBN 978-90-481-3030-6, pp. 1-209
- [3] José Duato, Sudhakar Yalamanchili, Lionel M. Ni: Interconnection networks - an engineering approach. IEEE 1997
- [4] L. Benini and G. D. Micheli, "Networks on Chips: A New SoC Paradigm," IEEE Computer, vol. 35, pp. 70–78, 2002.
- [5] T. D. Richardson, C. Nicopoulos, D. Park, V. Narayanan, X. Yuan, C. Das, and V. Degalahal, "A Hybrid SoC Interconnect with Dynamic TDMA-Based Transaction-Less Buses and On-Chip Networks," in Proceedings of the International Conference on VLSI Design, pp. 657–664, 2006.
- [6] R. Ho, K. W. Mai, and M. A. Horowitz, "The future of wires," in Proceedings of the IEEE, vol. 89, pp. 490–504, 2001.
- [7] International Technology Roadmap for Semiconductors (ITRS), <http://www.itrs.net/>, 2005 Edition.
- [8] M. B. Taylor, J. Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffman, P. Johnson, L. Jae-Wook, W. Lee, A. Ma, A. Saraf, M. Seneski, N. Shnidman, V. Strumpfen, M. Frank, S. Amarasinghe, and A. Agarwal, "The Raw microprocessor: a computational fabric for software circuits and general-purpose programs," IEEE Micro, vol. 22, pp. 25–35, 2002.
- [9] T. Simunic and S. Boyd, "Managing power consumption in networks on chips," in Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE), pp. 110–116, 2002.
- [10] P. Rickert, "Problems or opportunities? Beyond the 90nm frontier," ICCAD Keynote Address, 2004
- [11] Azimi, M.; Cherukuri, N.; Jayasimha, D.N.; Kumar, A.; Kundu, P.; Park, S.; Schoinas, I.; Vaidya, A. "Integration Challenges and Tradeoffs for Tera-scale Architectures." Intel Technology Journal. <http://www.intel.com/technology/itj/2007/v11i3/1-integration/1-abstract.htm> (August 2007).
- [12] Partha Pratim Pande; Grecu, C.; Jones, M.; Ivanov, A.; Saleh, R.; , "Performance evaluation and design trade-offs for network-on-chip interconnect architectures," IEEE Transactions on Computers, vol.54, no.8, pp. 1025- 1040, Aug. 2005 doi: 10.1109/TC.2005.134
- [13] A. Hemani, A. Jantsch, S. Kumar, A. Postula, J. Oberg, M. Millberg, and D. Lindqvist, "Network on chip: An architecture for billion transistor era," in Proceedings of the IEEE NorChip Conference, 2000.
- [14] P. Guerrier and A. Greiner, "A generic architecture for on-chip packet-switched interconnections," in Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE), pp. 250–256, 2000.
- [15] W.J. Dally and B. Towles, Route Packets, Not Wires: On-Chip Interconnection Networks, Proc. Design Automation Conf. (DAC), pp. 683-689, 2001.
- [16] Kumar, S.; Jantsch, A.; Soinenen, J.-P.; Forsell, M.; Millberg, M.; Oberg, J.; Tiensyrja, K.; Hemani, A.; , "A network on chip architecture and design methodology," 2002. Proceedings. IEEE Computer Society Annual Symposium on VLSI, vol., no., pp.105-112, 2002 doi: 10.1109/ISVLSI.2002.1016885
- [17] L. Benini and G. DeMicheli, Networks on Chips: A New SoC Paradigm, Computer, vol. 35, no. 1, pp. 70-78, Jan. 2002.

REFERENCES

- [18] R. Kumar, V. Zyuban, and D. M. Tullsen, "Interconnections in multi-core architectures: understanding mechanisms, overheads and scaling," in Proceedings of the 32nd International Symposium on Computer Architecture (ISCA), pp. 408–419, 2005.
- [19] S. Heo and K. Asanovic, "Replacing global wires with an on-chip network: a power analysis," in Proceedings of the 2005 International Symposium on Low Power Electronics and Design (ISLPED), pp. 369–374, 2005.
- [20] S. Li, L. S. Peh, and N. K. Jha, "PowerHerd: a distributed scheme for dynamically satisfying peak-power constraints in interconnection networks," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, pp. 92–110, 2006.
- [21] T. Dumitras, S. Kerner, and R. Marculescu, "Towards on-chip fault-tolerant communication," in Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 225–232, 2003.
- [22] R. Marculescu, "Networks-on-chip: the quest for on-chip fault-tolerant communication," in Proceedings of the IEEE Computer Society Annual Symposium on VLSI (ISVLSI), pp. 8–12, 2003.
- [23] A. Khakifirooz, Transport Enhancement Techniques for Nanoscale MOSFETs. PhD Thesis, MIT, Cambridge, MA, 2008.
- [24] A. Sheibanyrad, F. Pétrot, A. Jantsch. 2010. "3D-Integration for NoC-based SoC Architectures". Springer. ISBN 978-1-4419-7618-5
- [25] C. S. Tan, R. Gutmann, and R. Reif. 2008. "Wafer Level 3-D ICs Process Technology," Springer, ISBN 978-0-387-76532-7
- [26] D. Sylvester and C. Hu, Analytical Modeling and Characterization of Deep-Submicrometer Interconnect. Proceedings of the IEEE, 89(5), p. 634, 2001.
- [27] P.G. Emma, Is 3D Chip Technology the Next Growth Engine for Performance Improvement? IBM Journal of Research and Development, 52(6), p. 541, 2008.
- [28] Tezzaron: <http://www.tezzaron.com>
- [29] I. Loi, F. Angiolini, and L. Benini, "Supporting vertical links for 3d networks-on-chip: Toward an automated design and analysis flow," in International Conference on nano-networks, Catania, Italy, 2007.
- [30] B. Goplen and S. Sapatnekar, "Thermal Via Placement in 3D ICs", Proc. Intl. Symposium on Physical Design, p. 167, 2005.
- [31] J. Cong et al., "A Thermal-Driven Floorplanning Algorithm for 3D ICs", ICCAD, Nov. 2004.
- [32] W.-L. Hung et al., "Interconnect and Thermal-Aware Floorplanning for 3D Microprocessors", Proc. ISQED, March 2006.
- [33] S. K. Lim, "Physical Design for 3D System on Package", IEEE Design & Test of Computers, vol. 22, no. 6, pp. 532–539, 2005.
- [34] P. Zhou et al., "3D-STAF: Scalable Temperature and Leakage Aware Floorplanning for Three-Dimensional Integrated Circuits", ICCAD, Nov. 2007.
- [35] Sung Kyu Lim, "3D Circuit Design with Through-Silicon-Via: Challenges and Opportunities", IEEE Electronic Design Processes (EDP) Symposium Workshop, 2010.
- [36] International Technology Roadmap for Semiconductors (ITRS), <http://www.itrs.net/>, 2009 Edition.
- [37] W.J. Dally, B. Towles: Principles and Practices of Interconnection Networks. Morgan Kaufmann, 2004.
- [38] Healey, D. (2007). Design and Analysis of On-Chip Communication for Network-on-Chip Platforms. Unpublished doctoral dissertation, Royal Institute of Technology (KTH), Sweden.
- [39] Vasilis F. Pavlidis and Eby G. Friedman. 2009. Three-Dimensional Integrated Circuit Design. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. ISBN 0123743435 9780123743435

REFERENCES

- [40] Y. Xie, J. Cong, S. Sapatnekar. 2010."Three-Dimensional Integrated Circuit Design: EDA, Design and Microarchitectures (Integrated Circuits and Systems)". Springer. ISBN 978-1-4419-0784-4
- [41] A.Papanikolaou, D.Soudris, R. Radojcic. 2011. "Three Dimensional System Integration: IC Stacking Process and Design". Springer. ISBN 978-1-4419-0962-6
- [42] Feihui Li; Nicopoulos, C.; Richardson, T.; Yuan Xie; Narayanan, V.; Kandemir, M.; , "Design and Management of 3D Chip Multiprocessors Using Network-in-Memory,". 33rd International Symposium on Computer Architecture, 2006 (ISCA '06)., vol., no., pp.130-141, 0-0 doi: 10.1109/ISCA.2006.18
- [43] Pavlidis, V.F.; Friedman, E.G.; , "3-D Topologies for Networks-on-Chip,". IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.15, no.10, pp.1081-1090, Oct. 2007 doi: 10.1109/TVLSI.2007.893649
- [44] Feero, B.S.; Pande, P.P.; , "Networks-on-Chip in a Three-Dimensional Environment: A Performance Evaluation,". IEEE Transactions on Computers., vol.58, no.1, pp.32-45, Jan. 2009 doi: 10.1109/TC.2008.142
- [45] Seiculescu, C.; Murali, S.; Benini, L.; De Micheli, G.; , "SunFloor 3D: A tool for Networks On Chip topology synthesis for 3D systems on chips," Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE '09. , vol., no., pp.9-14, 20-24 April 2009 doi: 10.1109/DATE.2009.5090625
- [46] Shan Yan; Bill Lin, "Design of application-specific 3D Networks-on-Chip architectures," IEEE International Conference on Computer Design, 2008. ICCD 2008., vol., no., pp.142,149, 12-15 Oct. 2008 doi: 10.1109/ICCD.2008.4751853
- [47] Pingqiang Zhou; Ping-Hung Yuh; Sapatnekar, S.S., "Application-specific 3D Network-on-Chip design using simulated allocation," Design Automation Conference (ASP-DAC), 2010 15th Asia and South Pacific , vol., no., pp.517,522, 18-21 Jan. 2010 doi: 10.1109/ASPDAC.2010.5419830
- [48] Matsutani, H.; Koibuchi, M.; Amano, H.; , "Tightly-Coupled Multi-Layer Topologies for 3-D NoCs,". International Conference on Parallel Processing, 2007 (ICPP 2007)., vol., no., pp.75, 10-14 Sept. 2007 doi: 10.1109/ICPP.2007.79
- [49] Rahmani, A.-M.; Latif, K.; Vaddina, K.R.; Liljeberg, P.; Plosila, J.; Tenhunen, H.; , "Congestion aware, fault tolerant, and thermally efficient inter-layer communication scheme for hybrid NoC-bus 3D architectures,". 2011 Fifth IEEE/ACM International Symposium on Networks on Chip, (NoCS)., vol., no., pp.65-72, 1-4 May 2011
- [50] Chaochao Feng; Minxuan Zhang; Jinwen Li; Jiang Jiang; Zhonghai Lu; Jantsch, A.; , "A Low-Overhead Fault-Aware Deflection Routing Algorithm for 3D Network-on-Chip,". 2011 IEEE Computer Society Annual Symposium on VLSI, (ISVLSI)., vol., no., pp.19-24, 4-6 July 2011 doi: 10.1109/ISVLSI.2011.42
- [51] Xiaohang Wang; Palesi, M.; Mei Yang; Yingtao Jiang; Huang, M.C.; Peng Liu; , "Low latency and energy efficient multicasting schemes for 3D NoC-based SoCs,". 2011 IEEE/IFIP 19th International Conference on VLSI and System-on-Chip (VLSI-SoC)., vol., no., pp.337-342, 3-5 Oct. 2011 doi: 10.1109/VLSISoC.2011.6081604
- [52] Akbari, S.; Shafiee, A.; Fathy, M.; Berangi, R.; , "AFRA: A low cost high performance reliable routing for 3D mesh NoCs," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012., vol., no., pp.332-337, 12-16 March 2012 doi: 10.1109/DATE.2012.6176490
- [53] Chen, K.; Lin, S.; Hung, H.; Wu, A.; , "Topology-Aware Adaptive Routing for Non-Stationary Irregular Mesh in Throttled 3D NoC Systems,". IEEE Transactions on Parallel and Distributed Systems, vol.PP, no.99, pp.1, 0 doi: 10.1109/TPDS.2012.291

REFERENCES

- [54] Haoyuan Ying; Jaiswal, A.; Hofmann, K.; , "Deadlock-free routing algorithms for 3-dimension Networks-on-Chip with reduced vertical channel density topologies," International Conference on High Performance Computing and Simulation (HPCS), 2012, vol., no., pp.268-274, 2-6 July 2012 doi: 10.1109/HPCSim.2012.6266923
- [55] Bartzas, A., Skalis, N., Siozios, K., Soudris, D.: Exploration of alternative topologies for application-specific 3d networks-on-chip. In: Proc. of WASP 2007.
- [56] Fayez Gebali, Haytham Elmiligi, and Mohamed Watheq El-Kharashi. 2009. Networks-On-Chips: Theory and Practice (1st ed.). CRC Press, Inc., Boca Raton, FL, USA. ISBN 978-1-4200-7978-4
- [57] Weldezion, A.Y.; Grange, M.; Pamunuwa, D.; Zhonghai Lu; Jantsch, A.; Weerasekera, R.; Tenhunen, H.; , "Scalability of network-on-chip communication architecture for 3-D meshes," 3rd ACM/IEEE International Symposium on Networks-on-Chip, 2009. NoCS 2009., vol., no., pp.114-123, 10-13 May 2009
doi: 10.1109/NOCS.2009.5071459
- [58] Coskun, A.K.; Kahng, A.B.; Rosing, T.S.; , "Temperature- and Cost-Aware Design of 3D Multiprocessor Architectures," 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools, 2009. DSD '09., vol., no., pp.183-190, 27-29 Aug. 2009
doi: 10.1109/DSD.2009.233
- [59] Xu, T.C.; Liljeberg, P.; Tenhunen, H.; , "Optimal number and placement of Through Silicon Vias in 3D Network-on-Chip," IEEE 14th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2011, vol., no., pp.105-110, 13-15 April 2011
doi: 10.1109/DDECS.2011.5783057
- [60] Chia-I Chen; Bau-Cheng Lee; Juinn-Dar Huang; , "Architectural exploration of 3D FPGAs towards a better balance between area and delay," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011 , vol., no., pp.1-4, 14-18 March 2011
doi: 10.1109/DATE.2011.5763290
- [61] Healy, M.B.; Sung Kyu Lim; , "A novel TSV topology for many-tier 3D power-delivery networks," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011 , vol., no., pp.1-4, 14-18 March 2011
doi: 10.1109/DATE.2011.5763270
- [62] Jongman Kim, Chrysostomos Nicopoulos, Dongkook Park, Reetuparna Das, Yuan Xie, Vijaykrishnan Narayanan, Mazin S. Yousif, and Chita R. Das. 2007. A novel dimensionally-decomposed router for on-chip communication in 3D architectures. In Proceedings of the 34th annual international symposium on Computer architecture (ISCA '07). ACM, New York, NY, USA, 138-149. doi:10.1145/1250662.1250680
- [63] Dongkook Park; Eachempati, S.; Das, R.; Mishra, A.K.; Yuan Xie; Vijaykrishnan, N.; Das, C.R.; , "MIRA: A Multi-layered On-Chip Interconnect Router Architecture,". 35th International Symposium on Computer Architecture, 2008 (ISCA '08)., vol., no., pp.251-261, 21-25 June 2008
doi: 10.1109/ISCA.2008.13
- [64] Lafi, W.; Lattard, D.; Jerraya, A.; , "An efficient hierarchical router for large 3D NoCs," 2010 21st IEEE International Symposium on Rapid System Prototyping (RSP), vol., no., pp.1-5, 8-11 June 2010 doi: 10.1109/RSP.2010.5656418
- [65] Darve, F.; Sheibanyrad, A.; Vivet, P.; Petrot, F.; , "Physical Implementation of an Asynchronous 3D-NoC Router Using Serial Vertical Links," 2011 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), vol., no., pp.25-30, 4-6 July 2011
- [66] Latif, K.; Rahmani, A.-M.; Nigussie, E.; Tenhunen, H.; Seceleanu, T.; , "A Novel Topology-Independent Router Architecture to Enhance Reliability and Performance of Networks-on-Chip," IEEE International Symposium on Defect and Fault Tolerance in VLSI

REFERENCES

- and Nanotechnology Systems (DFT), 2011, vol., no., pp.454-462, 3-5 Oct. 2011 doi: 10.1109/DFT.2011.16
- [67] Rahmani, A.; Liljeberg, P.; Plosila, J.; Ka Lok Man; Youngmin Kim; Tenhunen, H.; , "Partial-LastZ: An optimized hybridization technique for 3D NoC architecture enabling adaptive inter-layer communication," International SoC Design Conference 2012 (ISOCC), vol., no., pp.281-284, 4-7 Nov. 2012 doi: 10.1109/ISOCC.2012.6407095
- [68] Dubois, F.; Sheibanyrad, A.; Pétrot, F.; Bahmani, M., "Elevator-First: A Deadlock-Free Distributed Routing Algorithm for Vertically Partially Connected 3D-NoCs," IEEE Transactions on Computers , vol.62, no.3, pp.609,615, March 2013 doi: 10.1109/TC.2011.239
- [69] M. Dehyadgari, M. Nickray, A. Afzali-kusha, Z. Navabi: Evaluation of Pseudo Adaptive XY Routing Using an Object Oriented Model for NOC. The 17th International Conference on Microelectronics, 13–15 December 2005.
- [70] Foroutan, S.; Thonnart, Y.; Hersemeule, R.; Jerraya, A., "An analytical method for evaluating Network-on-Chip performance," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010 , vol., no., pp.1629,1632, 8-12 March 2010 doi: 10.1109/DATE.2010.5457072
- [71] Foroutan, S.; Thonnart, Y.; Petrot, F., "An Iterative Computational Technique for Performance Evaluation of Networks-on-Chip," IEEE Transactions on Computers, vol.PP, no.99, pp.1,1, 0 doi: 10.1109/TC.2012.85
- [72] Bahmani, M.; Sheibanyrad, A.; Petrot, F.; Dubois, F.; Durante, P., "A 3D-NoC Router Implementation Exploiting Vertically-Partially-Connected Topologies," 2012 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), vol., no., pp.9,14, 19-21 Aug. 2012 doi: 10.1109/ISVLSI.2012.19
- [73] I. M. PANADES, A. GREINER, and A. SHEIBANYRAD. "A Low Cost Network-on-Chip with Guaranteed Service Well Suited to the GALS Approach". In Proc. the 1st Int'l Conf. and Workshop on Nano-Networks, pages 1–5, 2006.
- [74] Le Beux, S.; Trajkovic, J.; O'Connor, I.; Nicolescu, G.; Bois, G.; Paulin, P., "Optical Ring Network-on-Chip (ORNoC): Architecture and design methodology," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011 , vol., no., pp.1,6, 14-18 March 2011doi: 10.1109/DATE.2011.5763134
-